

**COMPUTATIONAL INTELLIGENCE
AND MODERN HEURISTICS**

**COMPUTATIONAL INTELLIGENCE
AND MODERN HEURISTICS**

Edited by
AL-DAHOUH ALI

Published by In-Teh

In-Teh

Olajnica 19/2, 32000 Vukovar, Croatia

Abstracting and non-profit use of the material is permitted with credit to the source. Statements and opinions expressed in the chapters are these of the individual contributors and not necessarily those of the editors or publisher. No responsibility is accepted for the accuracy of information contained in the published articles. Publisher assumes no responsibility liability for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained inside. After this work has been published by the In-Teh, authors have the right to republish it, in whole or part, in any publication of which they are an author or editor, and the make other personal use of the work.

© 2010 In-teh

www.intechweb.org

Additional copies can be obtained from:

publication@intechweb.org

First published February 2010

Printed in India

Technical Editor: Goran Bajac

Cover designed by Dino Smrekar

Computational Intelligence and Modern Heuristics,

Edited by Al-Dahoud Ali

p. cm.

ISBN 978-953-7619-28-2

Preface

The chapters of this book are collected mainly from the best selected papers that have been published in the 4th International conference on Information Technology ICIT 2009, that has been held in Al-Zaytoonah University/Jordan in the period 3-5/6/2009. The other chapters have been collected as related works to the book's topics.

“Heuristics are criteria, methods, or principles for deciding which among several alternative courses of action promises to be the most effective in order to achieve some goal - Pearl 1984

The term computational intelligence has become increasingly fuzzy, as the words “intelligent” and “smart” are used for everything from clever design of cell phones, appliances, computers, to pet robots, cars, and missiles. This collection of chapters will take its readers on a stunning voyage of computational intelligence heuristics research and applications.

Computational intelligence techniques, ranging from neural networks, fuzzy logic, via genetic algorithms to support vector machines, case based, neighborhood search techniques, ant colonies, and particle swarm optimization are effective approaches with applications where problem domain knowledge exists. Clearly the use of heuristic is one time honored form of an information based strategy to circumvent the learning process. Modern heuristics criteria, methods represent a set of principles that though may not guarantee, are in practice proven to lead to “good quality” solutions or methods for deciding which among several alternative courses of action promise to be the most effective in order to achieve a specified goal.

Collection of chapters of this book will elaborate different ideas in support of quantitative modeling heuristics on suite of applications including Computational Intelligence & Modern Heuristics in: Artificial Neural Network, Cryptography, Encryption, Dependability Evaluation, E-learning, GIS, Modeling, Optimization Problem, Security, Cryptosystems, Social process Design, Web, and Web Architectures.

Al-Dahoud Ali



Dr. Al-Dahoud, is an associated professor at Al-Zaytoonah University, Amman, Jordan. He took his High Diploma from FON University Belgrade 1986, PhD from La Sabianza1/Italy and Kiev Polytechnic/Ukraine, on 1996. He worked at Al-Zaytoonah University since 1996 until now. He worked as visiting professor in many universities in Jordan and Middle East, as supervisor of master and PhD degrees in computer science. He established the ICIT conference since 2003 and he is the program chair of ICIT until now. He was the Vice President of the IT committee in the ministry of youth/Jordan, 2005, 2006. Al-Dahoud was the General Chair of (ICITST-2008), June 23-28, 2008, Dublin, Ireland (www.icitst.org).

He has directed and led many projects sponsored by NUFFIC/Netherlands:

His hobby is conference organization, so he participates in the following conferences as general chair, program chair, session's organizer or in the publicity committee:

- ICITs, ICITST, ICITNS, DepCos, ICTA, ACITs, IMCL, WSEAS, and AICCSA

Journals Activities: Al-Dahoud worked as Editor in Chief or guest editor or in the Editorial board of the following Journals:

Journal of Digital Information Management, IAJIT, Journal of Computer Science, Int. J. Internet Technology and Secured Transactions, and UBICC.

He published many books and journal papers, and participated as keynote speaker in many conferences worldwide.

Contents

Preface	V
1. Services net modeling for dependability analysis Wojciech Zamojski and Tomasz Walkowiak	001
2. Service based information systems analysis using task-level simulator Tomasz Walkowiak	017
3. Modelling equipment deterioration vs. maintenance policy in dependability analysis Jarosław Sugier and George J. Anders	029
4. Computational Experience in Methods for Finding Tight Lower Bounds for the Sparse Travelling Salesman Problem Fredrick Mtenzi	043
5. Modelling Access Control with Dynamic Role Binding Al-Dahoud Ali and Dr.K.Chitra	061
6. On the Accuracy of a Stewart Platform: Modelling and Experimental Validation Mircea Neagoie, Dorin Diaconescu, Codruta Jaliu, Sergiu-Dan Stan, Nadia Cretescu and Radu Saulescu	075
7. Effective knowledge acquisition by means of teaching strategies Marek Woda	099
8. Measuring Customer Service Satisfactions Using Fuzzy Artificial Neural Network with Two-phase Genetic Algorithm M. Reza Mashinchi and Ali Selamat	107
9. A Variation of Particle Swarm Optimization for Training of Artificial Neural Networks Masood Zamani and Alireza Sadeghian	131
10. Resilient Back Propagation Algorithm for Breast Biopsy Classification Based on Artificial Neural Networks Fawzi M. Al-Naima and Ali H. Al-Timemy	145
11. SIMD Architecture Approach to Artificial Neural Networks Realisation Jacek Mazurkiewicz	159

12. Smart RFID Security, Privacy and Authentication Mouza A. Bani Shemali, Chan Yeob Yeun and Mohamed Jamal Zemerly	175
13. Security and Privacy of Intelligent VANETs Mahmoud Al-Qutayri, Chan Yeun and Faisal Al-Hawi	191
14. New Classification of Existing Stream Ciphers Khaled Suwais and Azman Samsudin	219
15. Intelligent Exploitation of Cooperative Client-Proxy Caches in a Web Caching Hybrid Architecture Maha Saleh El Oneis, Mohamed Jamal Zemerly and Hassan Barada	241
16. Smart Web Based Programming Contests Management Tool Ahmed Bentiba, Mohamed J. Zemerly and Mohamed Al Mansoori	255
17. Heuristics of social process design Gilbert Ahamer	265
18. Heuristics and pattern recognition in complex geo-referenced systems Gilbert Ahamer, Adrijana Car, Robert Marschallinger, Gudrun Wallentin and Fritz Zobl	299
19. Complexity of Instances for Combinatorial Optimization Problems Jorge A. Ruiz-Vanoye, Ocotlán Díaz-Parra, Joaquín Pérez-Ortega, Rodolfo A. Pazos R. Gerardo Reyes Salgado and Juan Javier González-Barbosa	319
20. Dependability Evaluation Based on System Monitoring Janusz Sosnowski and Marcin Król	331

Services net modeling for dependability analysis

Wojciech Zamojski and Tomasz Walkowiak
Wroclaw University of Technology
Poland

1. Introduction

Network technologies are being developed for many years. Most of large technical systems could be seen as a kind of network, for example: information, transport or electricity distribution systems. Networks are modelled as directed graphs with nodes, in which commodities and information media are being processed, and arcs as communication links (telecommunication channels, roads, pipelines, conveyors, etc.) for media transportation. Resources of networks could be divided into two classes: services (functionality resources) and technical infrastructures (hardware and software resources).

We propose to analyse the network system from the functional and user point of view, focusing on business service realized by a network system (Gold et al., 2004). Users of the network system realise some tasks in the system (for example: send a parcel in the transport system or buy a ticket in the internet ticket office). We assume that the main goal, taken into consideration during design and operation, of the network system is to fulfil the user requirements. Which could be seen as some quantitative and qualitative parameters of user tasks.

Network services and technical resources are engaged for task realization and each task needs a fixed list of services which are processed on the base of whole network technical infrastructure or on its part. Different services may be realized on the same technical resources and the same services may be realized on different sets of technical resources. Of course with different values of performance and reliability parameters. The last statement is essential when tasks are realized in the real network system surrounded by unfriendly environment that may be a source of threads and even intentional attacks. Moreover, the real networks are build of unreliable software and hardware components as well.

In (Avižienis et al., 2000) authors described basic set of dependability attributes (i.e. availability, reliability, safety, confidentiality, integrity and maintainability). This is a base of defining different dependability metrics used in dependability analysis of computer systems and networks. In this paper we would like to focus on more functional approach metrics which could be used by the operator of the network system. Therefore, we consider dependability of networks as a property of the networks to reliable process of user tasks, that is mean the tasks have to perform not only without faults but more with demanded performance parameters and according to the planned schedule.

We propose to concentrate the dependability analyse of the networks on fulfilling the user requirements. Therefore, it should take into consideration following aspects:

- specification of the user requirements described by task demands, for example certainty of results, confidentiality, desired time parameters etc.,
- functional and performance properties of the networks and their components,
- reliable properties of the network technical infrastructure that means reliable properties of the network structure and its components considered as a source of failures and faults which influence the task processing,
- process of faults management,
- threads in the network environment,
- measures and methods which are planned or build-in the network for elimination or limitation of faults, failures and attacks consequences; reconfiguration of the network is a good example of such methods,
- applied maintenance policies in the considered network.

As a consequence, a services network is considered as a dynamical structure with many streams of events generated by realized tasks, used services and resources, applied maintenance policies, manager decisions etc. Some network events are independent but other ones are direct consequences of previously history of the network life. Generally, event streams created by a real network are a mix of deterministic and stochastic streams which are strongly tied together by a network choreography. Modelling of this kind of systems is a hard problem for system designers, constructors and maintenance organizers, and for mathematicians, too. It is worth to point out some achievements in computer science area such as Service Oriented Architecture (Gold et al., 2004; Josuttis, 2007) or Business Oriented Architecture (Zhu & Zhang, 2006) and a lot of languages for network description on a system choreography level, for example WS-CDL (Yang et al., 2006), or a technical infrastructure level, for example SDL (Aime et al., 2007). These propositions are useful for analysis of a network from the designer point of view and they may be supported by simulation tools, for example modified SSF.Net simulator (Zyla & Caban, 2008), but it is difficult to find a computer tools which are combination of language models and Monte Carlo (Fishman, 1996) based simulators.

The chapter presents a step to a creation of a verbal and formal model of a net of services. It presents a generic approach to modelling performability (performance and reliability) properties of the services net. The Petri Nets will be used for the task realization process modelling. Moreover, an example of service net – the discrete transport system analysed by an event-driven simulator is presented.

2. Service network – overview

We can distinguish three main elements of any network system: users, services and technical resources. As it is presented in the Figure 1 users are generating tasks which are being realized by the network system. The task to be realized requires some services presented in the system. A realization of the network service needs a defined set of technical resources. In a case when any resource component of this set is in a state "out of order" or "busy" then the network service may wait until a moment when the resource component

returns to a state "available" or the service may try to create other configuration on the base of available technical resources.

Therefore, following problems should be taken into consideration:

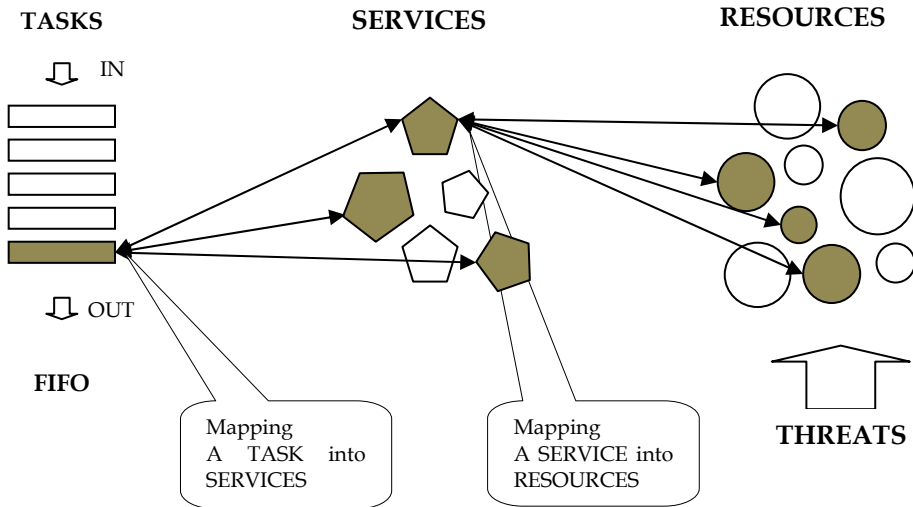


Fig. 1. Task mapping on business services and technical resources

- description and mapping a service net on existed net resources for each moment of its using;
- a prognoses process of the service net behaviour in a real life conditions – definition and selection of measures;
- finding relations between measures/criteria and functional, performance and reliability parameters of the service net;
- evaluation methods of choose measures of the service net;
- decision process of maintenance organization - decision steps as a reaction on appeared events, specially on threats;
- definition of measures and criteria of decision steps - risk of threats, and evaluation of decision risk and its cost.

An illustration of problems connected with functional – dependability modelling of services networks is shown in Figure 2.

3. Functional – dependability models

The *ST model* (*State - Transition model*) is the most popular and useful methodology used in modelling of systems.

The system is considered as a union of its hardware, management system and involved personnel (administrators, users, support services etc.), so the system states depend on the states of all these elements. The system transitions are consequences of events connected

with execution of system tasks and jobs, system faults and system reactions to them, incidents, attacks and system responses etc., i.e. system events are observable occurrences which change states of the system.

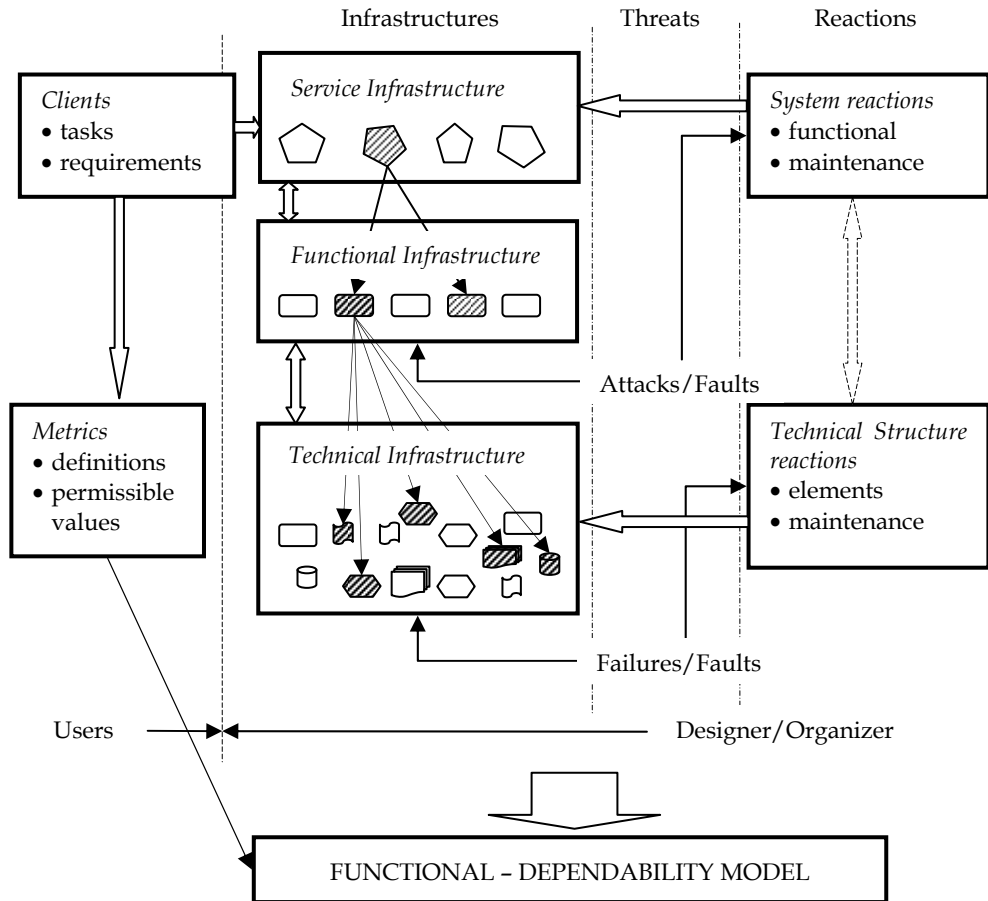


Fig. 2. Basic terms and a functional - dependability model of a services network (Zamojski, 2009)

The functional - reliability model (Zamojski, 2005) of computer system S_C is a configuration of hardware H , software SP , men M , management system (operating system) MS , tasks (functions) J and system events E_S

$$S_C \subset H \times SP \times J \times M \times MS \times E_S \quad (1)$$

The system events includes those connected with tasks realization, occurrence of incidents (faults, viruses, and attacks) and system reactions to them (hardware and information

renewals). The system events are very often described by their time parameters which are collected in so called a *chronicle* of the system.

A functional configuration $S_C^{(i)}$ of the computer system is a set of hardware and software resources that are allocated to realize i -th task $j^{(i)}$;

$$(j^{(i)} \subset J) \Rightarrow (S_C^{(i)} \subset S_C) \tag{2}$$

and

$$S_C^{(i)} \subseteq H^{(i)} \times SP^{(i)} \times j^{(i)} \times M^{(i)} \times MS^{(i)} \times E_S^{(i)} \tag{3}$$

where superscript (i) fix subsets of system resources needed for execution i -th task.

A functional – reliability model in the system engineering is regarded as a structured representation of the functions, activities or processes, and events generated inside of the considered system and/or by its surroundings. The system events may be divided into two main classes: functional events and reliable (together with maintenance) events. In practice this classification is very often difficult to be made because a system reaction on an event may involve a lot of functional or/and maintenance reactions. Therefore, it is better to create one common class of functional–reliable events, so called *performability* events (Zamojski & Caban, 2006). Because of these reasons considered model of services network will be called *performability* model or *functional-dependability* model (Zamojski & Caban, 2007).

If the functional – reliability model is built as the ST model then the set of the system states is determined by the states of all resources involved in tasks realized at the moment. The system resource allocations are dynamic, modified due to the incoming tasks, occurring incidents and system reactions (especially reconfiguration).

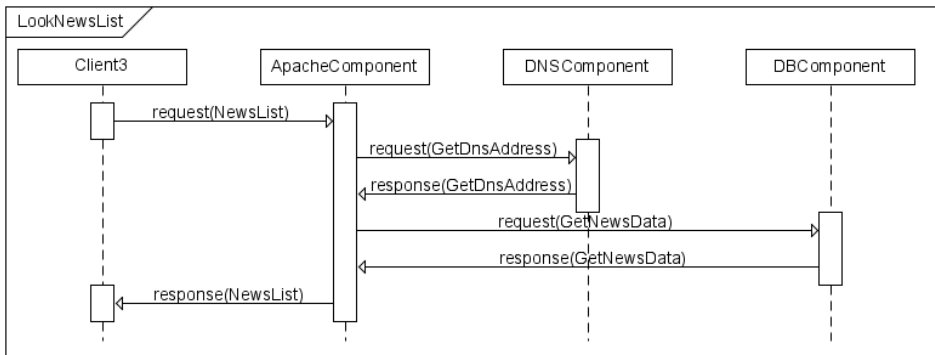


Fig. 3. Exemplar choreography

4. Formal model of a service net

4.1 A service net

A *services net* is a system of business services that are necessary for user (clients) tasks realization process. The services net are built on the bases of technical infrastructure

(*technological resources*) and *technological services* which are involved into a task realization process according to decisions of a management system. The task realization process may include many sequences of services, functions and operations which are using assignment network resources - in the computer science this process of assignments and realization steps is called as a *choreography*. An example of choreography for web service is presented in Figure 3.

The functional - dependability model of a services network has to consider specificity of the network: nodes and communication channels, the ability of dynamic changes of network traffic (routing) and reconfiguration, and all other tasks realized by the network.

The service network could be defined as a tuple:

$$SNet = \langle J, BS, TR, MS, C \rangle, \quad (4)$$

where:

- $J = \{J^{(i)}; i = 1, 2, \dots\}$ - a set of tasks generated by users and realized by the service network,
- $BS = \{BS^{(b)}; b = 1, 2, \dots\}$ - a set of services which are available in the considered network,
- $TR = \{TR^{(r)}; r = 1, 2, \dots\}$ - technical infrastructure of the network which consists of technical resources as machines/servers, communication links etc,
- MS - management system (for example - operating system),
- $C = \{c_t; t = 1, 2, \dots\}$ - a network chronicle, defined by a set of all essential moments in a "life" of the network.

4.2 Tasks

The task $J^{(i)}$ is understood as a sequence of actions and works performed by services network in a purpose to obtain desirable results in accordance with initially predefined time schedule and data results. In this way a single task $J^{(i)} = \langle J_{IN}^{(i)}, J_{OUT}^{(i)} \rangle$ may be defined as an ordered pair of so called *input task* $J_{IN}^{(i)}$, which is described by the input parameters (postulated results and prognosis time schedule) and the corresponding *output task* $J_{OUT}^{(i)}$ (real results and real time schedule).

The input task is define as the triple:

$$J_{IN}^{(i)} = \langle R_p^{(i)}, A^{(i)}, C_p^{(i)} \rangle, \quad (5)$$

where:

- $R_p^{(i)}$ - postulated results of the i -th task execution,
- $C_p^{(i)}$ - postulated chronicle of the task realization,
- $A^{(i)} = A^{(i)}(R_p^{(i)}, C_p^{(i)})$ - a sequence of actions and works necessary to obtain postulated results in planned time.

The $A^{(i)}$ may be described by a flowchart of actions and works, and its realization depends on an availability of network services and technical resources.

The output task is define as the pair:

$$J_{OUT}^{(i)} = \langle R_{real}^{(i)}, C_{real}^{(i)} \rangle, \quad (6)$$

where:

- $R_{real}^{(i)}$ - real results of the i -th task execution,
- $C_{real}^{(i)}$ - real chronicle of the task realization.

The postulated results and chronicles are defined with assumed tolerance intervals ($\underline{R}_p^{(i)} \leq R_p^{(i)} \leq \overline{R}_p^{(i)}$ and $\underline{C}_p^{(i)} \leq C_p^{(i)} \leq \overline{C}_p^{(i)}$) and when the real results and chronicles are inside the intervals ($R_{real}^{(i)} \in [\underline{R}_p^{(i)}, \overline{R}_p^{(i)}]$ and $C_{real}^{(i)} \in [\underline{C}_p^{(i)}, \overline{C}_p^{(i)}]$) then the task is assumed to be correctly realised.

4.3 Services

The term service is understood as a discretely defined set of contiguously cooperating autonomous business or technical functionalities. Of course, a special mechanism to enable an access to one or more businesses and functionalities should be implemented in the system. The access is provided by a prescribed interface and is monitored and controlled according to constraints and policies as specified by the service description¹.

The service $BS^{(b)}$ is defined as a sequence of activities described by a set of capabilities (functionalities) $\{F_k^{(b)}, k = 1, 2, \dots\}$, a set of demanded input parameters of data and/or media $BS_{IN}^{(b)}$ and a set of output parameters $BS_{OUT}^{(b)}$:

$$BS^{(b)} = \langle \{F_k^{(b)}; k = 1, 2, \dots\}, BS_{IN}^{(b)}, BS_{OUT}^{(b)} \rangle. \quad (7)$$

Because the services have to cooperate with other services than protocols and interfaces between services and/or individual activities are crucial problems which have a big impact on the definitions of the services and on processes of their execution.

A service may be realized on the base of a few separated sets of functionalities $\{F_{k1}^{(b)}, k1 = 1, 2, \dots\}, \{F_{k2}^{(b)}, k2 = 1, 2, \dots\}$... with different costs which are the consequences of using different network resources.

4.4 Technical infrastructures

Hardware is considered as a set of hardware resources (devices and communication channels) which are described by their technical, performance, reliability and maintenance parameters. The system software is described in the same way.

¹ OASIS Organization for the Advancement of Structured Information Standards Home Page. <http://www.oasis-open.org/home/index.php>

4.5 Management system

The management system of service network allocates the services and network resources to realized tasks, checks the efficient states of the services network, performs suitable actions to locate faults, attacks or viruses and minimize their negative effects. Generally the management system has two main functionalities:

- monitoring of network states and controlling of services and resources,
- creating and implementing maintenance policies which ought to be adequate network reactions on concrete events/accidents. In many critical situations a team of men and the management system have to cooperate in looking for adequate counter-measures, for instance in case of a heavy attack or a new virus.

The maintenance policy is based on two main concepts: detection of unfriendly events (attacks, faults, failures) and network responses to them. In general the network responses incorporate the following procedures:

- detection of incidents and identification of them,
- isolation of damaged network resources in order to limit proliferation of incident consequences,
- renewal of damaged services, processes and resources.

It is hard to predict all possible events (for example all new demands for a task realization) or incidents (for example failures, faults, attacks or an end of a renewal procedure) in the services network, especially it is not possible to predict all possible attacks or men faults, so system reactions are very often "improvised" by the management system, by its administrator staff or even by expert panels specially created to find a solution for the existing situation. The time, needed for the renewal, depends on the incident that has occurred, the system resources that are available and the renewal policy that is applied. The renewal policy is formulated on the basis of the required levels of system dependability and on the economical conditions (first of all, the cost of downtime and cost of lost achievements) (Zamojski & Caban, 2006; Zamojski & Caban, 2007).

Maintenance policy is based on maintenance rules that are understood as chains of decisions about allocation of services and network resources (hardware, software, information and service staff) that are undertaken to keep the system operational after an incident. These rules are very often connected with small fragments of the system, for example; replacement of a machine (a processor) or communication links. These local operations may have impact on the whole network, e.g. if a communication channel is down for a few minutes, then rates of medium (data) traffic of the network may violently change (Zamojski & Caban, 2007).

4.6 Chronicles

The set of system events is created by events connected with tasks realization, incidents occurrence (faults, viruses, and attacks) and system reactions (hardware and information renewals).

4.7 A process of the task realization

The task realization process is supported by two-level decision procedures connected with selection and allocation of the network functionalities and technical resources. There are two levels of decision process: services management and resource management. The first level of decision procedure is connected with selection suitable services and creation a task configuration. Functional and performance task demands are the base for suitable services choosing from all possible network services. The goal of the second level of the decision process is to find needed components of the network infrastructure for each service execution and the next allocate them on the base their availability to the service configuration. If any component of technical infrastructure is not ready to support the service configuration then allocation process of network infrastructure is repeated. If the management system could not create the service configuration then the service management process is started again and other task configuration may be appointed. These two decision processes are working in a loop which is started up as a reaction on network events and accidents. On the beginning of a task realization procedure the task $J_{IN}^{(i)}$ is mapped on the network services and a subset of services $BS_s^{(i)}$ necessary for the task realization according to its postulated parameters is created; $J_{IN}^{(i)} \rightarrow BS_s^{(i)}$. Next, a demand of technical resources for each service realization is fixed: $BS_s^{(i)} \rightarrow R_n^{(i,s)}$. In a real services network the same task is very often realized on the base of various service subsets and the same service may involved different technical resources. Of course, this possible diversity of task realization is connected with the flowcharts $A^{(i)}$ and the availability of network resources is checking for each service. In this way a few task configurations service configurations, additionally described by appropriately defined cost parameters, may be fund for the i -th task realization.

5. The Petri net model

Petri Nets (Zhou & Kurapati, 1999) are a powerful and often used modelling tool. They allow to represent two aspects of a modelled system static and dynamic (thanks to the token evolution). A common definition of the Petri net is formulating as a triple:

$$PN = \langle P, T, A \rangle \quad (8)$$

where:

- P - set of places that represent deterministic states of processes, tasks, services, resources etc. of the considered system. The places are often complemented by tokens that are modeled abilities of these places.
- T - set of transitions that represent net events characterized by conditions necessary to come them into firing. The transitions are often described by firing time and other probabilistic characteristics etc.
- A - set of arches (*directed* and *inhibited*) that models routes on which events represented by tokens are passed by the net.

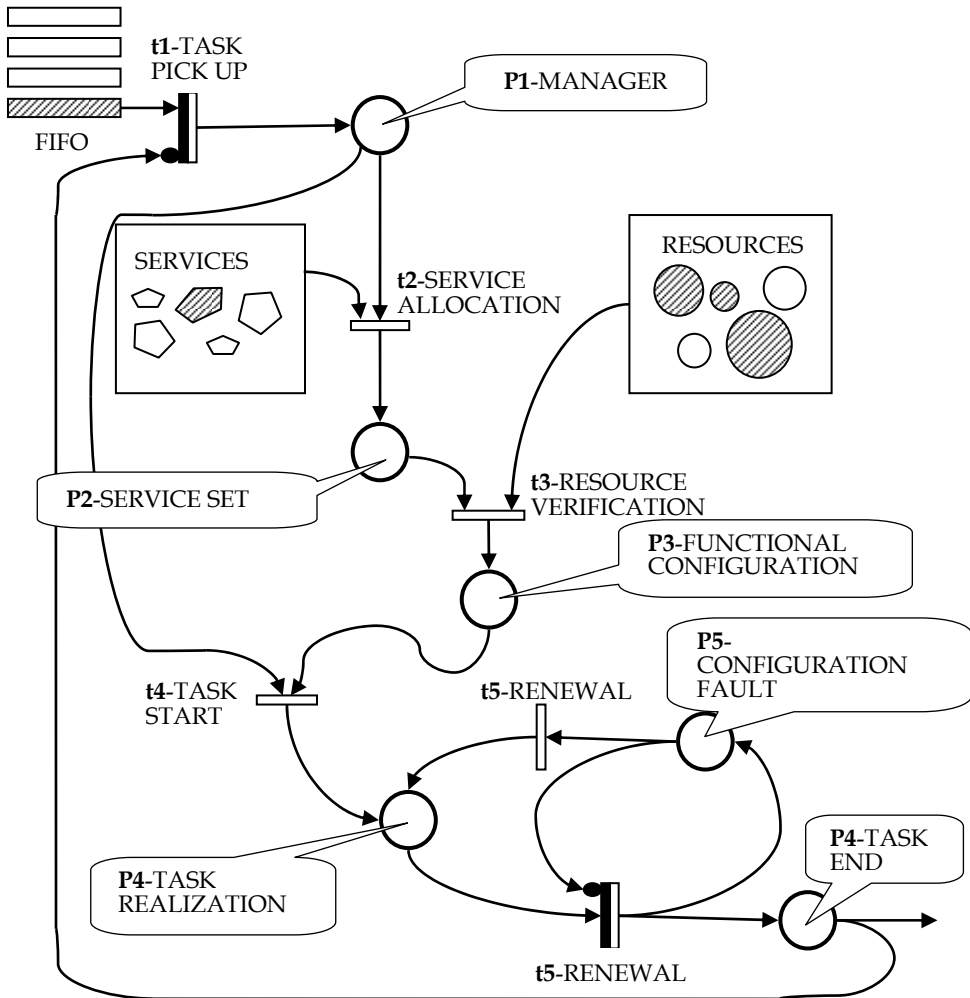


Fig. 4. The Petri net model of a task realization in a services network

A state of the net, described by *marking* (tokens localization in the places) represents sufficient conditions for arising new events of a net's life. Net's events may be divided into many classes, for example functional, reliable or maintenance events, deterministic or probabilistic ones etc. The mention classification depends on assumed criteria.

The Petri net model of the i^{th} task realization ($J^{(i)}$) is shown in the Figure 4. It is assumed the input task ($J_{IN}^{(i)}$) is taken from the stack of waiting tasks (transition $t1$ and its firing time $\tau_{T1}^{(i)}$). The choice of the task may be based on the strategy FIFO (as it is illustrated on the Figure 2) and it is conditioned by ending of previously task (the transition $t1$ is guarded by inhibited arc from the place $P6$ (end of the task)). The place $P1$ represents the management

process of mapping the input task into a set of necessary services ($BS^{(b)}$) and when the services are ready then the transition $t2$ is fired (time $\tau_{T2}^{(i)}$). After checking if the chosen services may be activated on the base of needed efficient technical resources then a functional configuration of the task (place $P3$) is created (transition $t3$ with time $\tau_{T3}^{(i)}$) and at this moment the manager may take a decision about start of the task process realization (transition $t4$).

There is a build-in system of monitoring and detection of unfriendly accidents like faults and failures (place $P5$). When such unfriendly accident is discovered then a renewal process of the functional configuration is started (transition $t5$ and renewal time $\tau_{T5}^{(i)}$) and the task realization process is broken (the inhibited input of the transition $t6$) till the end of renewal operations.

The firing process of each transition is described by conditions (tokens in input places for the transition) which may occur with probabilities, for example a probability of a machine failure, and time duration of transition firing may be a probabilistic function, too. Of course a transition may be many times fired during a task realization, because net events may need to repeat bigger or smaller loops of the net. The Petri net model shown in the Figure 4 is reduced and presented only to show the main idea of the proposed modelling method which may be useful for evaluation of dependability measures of services networks.

Real time of the i^{th} task realization $T_{J_{real}}^{(i)}$ that is modelled as a stochastic timed Petri net with k transitions and l loops and sub loops may be evaluated as:

$$T_{J_{real}}^{(i)} \cong \sum_{l \in L} \Pr\{e_l^{(i)} = 1\} \left[\sum_k \Pr\{f_{T_{k,l}}^{(i)} = 1\} \tau_{T_{k,l}} \right], \quad (9)$$

where:

- $e_l^{(i)} = 1$ - an event (for example, a new task, an allocation a technical resource to the i -th task, an end of a renewal process etc.) which is started a loop or a sub loop in the Petri net model ascribed to the i th task realisation,
- $f_{T_{k,l}}^{(i)} = 1$ - an event; the k transition is fired during l loop connected with the i -th task realization.

Such dependability measures as a probability that the real time duration of the i -th task may be defined and evaluated on the base of the Petri net models as:

$$M_{Depend}^{(i)}(J_{IN}^{(i)}) = \Pr\{T_{J_{real}}^{(i)} \leq T_{J_{OUT}}^{(i)}\}. \quad (10)$$

6. Discrete transport system – service net case study

An example of service net could be a DTSCNTT - Discrete Transport System with Central Node and Time-Table (Walkowiak et al., 2007). This is a simplified case of the Polish Post transport system.

Following the definition (4) each elements of service net could be described as follows.

The business service (*BS*) provided the Polish Post and therefore DTSNTT service net is the delivery of mails. The technical infrastructure (*TR*) consists of a set of nodes placed in different geographical locations and set of vehicles and timetable. There are bidirectional routes between nodes marked by lines. There is distinguished one node called central node. Mails are distributed among nodes by vehicles.

Each vehicle is described by following functional and reliability parameters: mean speed of a journey, capacity – number of containers which can be loaded, reliability function and time of vehicle maintenance.

Management system (*MS*) is defined by time table since vehicles distributing mails among system nodes operate according to the time-table exactly as city buses or intercity coaches. The time-table consists of a set of routes (sequence of nodes starting and ending in the central node, time of approaching each node in the route and the recommended size of a vehicle). The number of used vehicle, or the capacity of vehicles does not depend on temporary situation described by number of transportation tasks or by the task amount for example. It means that it is possible to realize the journey by completely empty vehicle or the vehicle cannot load the available amount of commodity (the vehicle is too small). Time-table is a fixed element of the system in observable time horizon, but it is possible to use different time-tables for different seasons or months of the year.

To reduce the complexity of the model we have decided to model the containers not separate mails (Walkowiak & Mazurkiewicz, 2009). Therefore, the tasks (*J*) of sending mails is modelled as a random process of containers generation. Each generated container has a destination address. The central node is the destination address for all containers generated in the ordinary nodes. Where containers addressed to in any ordinary nodes are generated in the central node. The generation of containers is described by Poisson process. In case of central node there are separate processes for each ordinary node. Whereas, for ordinary nodes there is one process, since commodities are transported from ordinary nodes to the central node or in opposite direction. Postulated result of any task is to transport a container to the destination node within a given time limit.

The process of any task realization could be described as follows. The container is generated in some node at a given time (according to Poisson process) and stored in the node waiting for the vehicle to be transported to the destination node. Each day a given time-table is realized, it means that at a time given by the time table a vehicle, selected randomly from vehicles available in the central node, starts from central node and is loaded with containers addressed to each ordinary nodes included in a given route. The loading is done in a service point. This is done in a proportional way. Since the number of service points is limited (parameter of the central node) and loading takes some time there is no free service point vehicles has to wait in a queue. After loading the vehicle goes to a given ordinary node - it takes some time according to vehicle speed - random process and road length. After approaching the ordinary node the vehicle is waiting in an input queue if there is any other vehicle being loaded/unloaded at the same time. The containers addressed to given node are unloaded and empty space in the vehicle is filled by containers addressed to a central node. The operation is repeated in each node on the route and finally the vehicle is approaching the central node when is fully unloaded and after it is available for the next route. The process of vehicle operation could be stopped at any moment due to a failure (described by a random process). After the failure, the vehicle waits for a maintenance crew

(if it is not available due to repairing other vehicles), is being repaired (random time) and after it continues its journey (Walkowiak & Mazurkiewicz, 2009).

As suggested in the introduction the simulator tool for analysing DTSCNTT service net was developed. The tool was adopting the event simulation approach, which is based on a idea of event, which could be described by time of event occurring, type of event (in case of DTSCNTT it could be a vehicle failure) and element or set of elements of the system on which event has its influence. The simulation is done by analyzing a queue of event (sorted by time of event occurring) while updating the states of system elements according to rules related to a proper type of an event. (Walkowiak et al., 2007)

We proposed for the case study analysis an exemplar DTSCNTT based on Polish Post regional centre in Wroclaw. We have modelled a system consisting of one central node (Wroclaw regional centre) and twenty two other nodes - cities where there are local post distribution points in Dolny Slask Province. The length of roads were set according to real road distances between cities used in the analyzed case study. The intensity of generation of containers for all destinations were set to 4,16 per hour in each direction giving in average 4400 containers to be transported each day. The vehicles speed was modelled by Gaussian distribution with 50 km/h of mean value and 5 km/h of standard deviation. The average loading time was equal to 5 minutes. There were two types of vehicles: with capacity of 10 and 15 containers. The MTF of each vehicle was set to 2000. The average repair time was set to 5h (Gaussian distribution). (Walkowiak & Mazurkiewicz, 2009)

The simulation time was set to 100 days and each simulation was repeated 10.000 times. We have calculated the dependability measure defined by (10), the probability that the duration time of a task (delivery of some container) will be longer then a given time limit using Monte-Carlo approach (Fishman, 1996). The achieved results are presented in Figure 5.

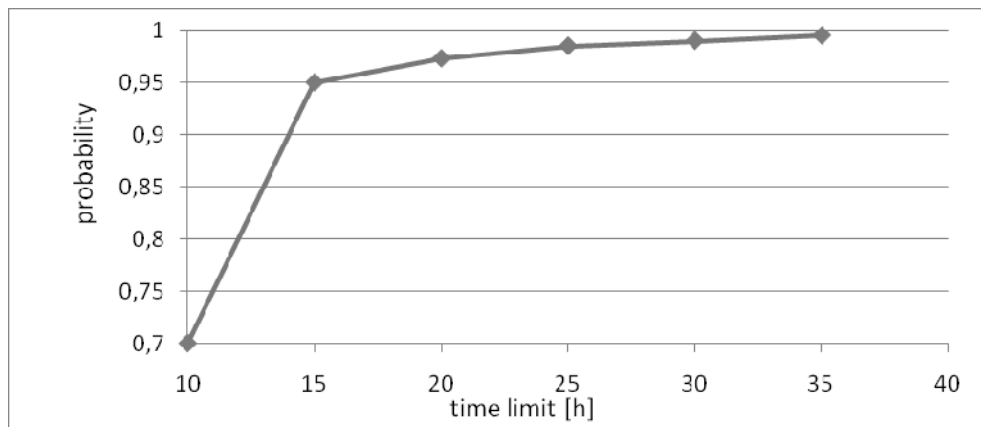


Fig. 5. The probability of containers to be transported within a given limit time

7. Conclusion

We have given a verbal and formal model of a service net. The formal model consists of a tuple mathematical model and the Petri Nets one. We hope that the proposed Petri net model will be very useful in the synthesis process of the service net. Of course there are a lot

problems with building the Petri net model of the real services net in which exist a large number of services and technical resources that are mapped to many concurrent realized tasks. We have also presented an exemplar case study of service net a discrete transport system service net – a simplified case of Polish Post transport system. It was analysed by a usage of a discrete transport system simulator.

We plan to develop a simulation tool for a generic service nets with a functionality similar to presented discrete transport system simulator or BS.SSF simulator (Walkowiak, 2009) together with graphical tool for modelling and simulation. We also plan to use high level languages like for examples Business Process Modeling Notation (White & Miers 2008) for a graphical representation for specifying business processes in a workflow. We hope that it could be possible to map BPMN into a Petri net model or a general purpose service net simulator allowing to perform a service net dependability analysis.

8. References

- Aime, M.; Atzeni, A.; Pomi, P. (2007). Ambra - Automated Model-Based Risk Analysis, *Proceedings of the 3rd International Workshop on Quality of Protection*, pp. 43-48, Alexandria, ACM, New York
- Avižienis, A.; Laprie, J.; Randell, B. (2000). Fundamental Concepts of Dependability. *Proceedings of 3rd Information Survivability Workshop*, Boston
- Fishman, G. (1996). *Monte Carlo: Concepts, Algorithms, and Applications*, Springer-Verlag, New York
- Gold, N.; Knight, C.; Mohan, A.; Munro, M. (2004). Understanding service-oriented software. *IEEE Software*, Vol. 21, 71–77
- Josuttis, N. (2007). *SOA in Practice: The Art of Distributed System Design*, O'Reilly
- Walkowiak, T. (2009). Information systems performance analysis using task-level simulator, *Proceedings of International Conference on Dependability of Computer Systems*, pp. 218-225, Brunow, IEEE Computer Society Press, Los Alamitos
- Walkowiak T.; Mazurkiewicz, J. (2009). Analysis of critical situations in discrete transport systems, *Proceedings of International Conference on Dependability of Computer Systems*, pp. 364-371, Brunow, IEEE Computer Society Press, Los Alamitos
- Walkowiak, T.; Mazurkiewicz, J.; Kaplon, K. (2007). Functional analysis of discrete transport system realized by SSF simulation tool. *Advances simulation of systems. Proceedings of the XXIXth International Autumn Colloquium*, pp. 103-108, Sv. Hostyn, MARQ, Ostrava
- White, S. A Miers, D. (2008). *BPMN Modeling and Reference Guide*, Future Strategies Inc., Lighthouse Pt
- Yang, H.; Zhao, X.; Qiu, Z.; Pu, G.; Wang, S. (2006). A Formal Model for Web Service Choreography Description Language (WS-CDL). *Proceedings of the IEEE international Conference on Web Services*, IEEE Computer Society, Washington
- Zamojski, W. (2005). Functional-reliability model of computer-human system. *Computer engineering*, pp. 278-297, Eds. Wojciech Zamojski, WKL, Warszawa (in Polish)
- Zamojski, W. (2009). Dependability of services networks. *Proceedings of the Third Summer Safety and Reliability Seminars*, pp. 387-396, Gdnask-Sopot, Polish Safety and Reliability Association, Gdansk

- Zamojski W.; Caban D. (2006). Introduction to the dependability modelling of computer systems. *Proceedings of International Conference on Dependability of Computer Systems*, pp. 100 – 109, Szklarska Poreba, IEEE Computer Society Press, Los Alamitos
- Zamojski, W.; Caban, D. (2007). Maintenance policy of a network with traffic reconfiguration. *Proceedings of International Conference on Dependability of Computer Systems*, pp. 213 – 220, Szklarska Poreba, IEEE Computer Society Press, Los Alamitos
- Zhu, J.; Zhang, L. Z. (2006). A Sandwich Model for Business Integration in BOA (Business Oriented Architecture). *Proceedings of the 2006 IEEE Asia-Pacific Conference on Services Computing*, pp. 305-310, IEEE Computer Society, Washington
- Zhou, M.; Kurapati, V (1999) *Modeling, Simulation, & Control of Flexible Manufacturing Systems: A Petri Net Approach*. World Scientific Publishing
- Zyla, M.; Caban, D. (2008). Dependability Analysis of SOA systems. *Proceedings of International Conference on Dependability of Computer Systems*, pp. 301-306, Szklarska Poreba, IEEE Computer Society Press, Los Alamitos

Service based information systems analysis using task-level simulator

Tomasz Walkowiak
Wroclaw University of Technology
Poland

1. Introduction

Complex information systems (CIS) are nowadays the core of a large number of companies. And therefore, there is a large need to analyze various system configuration and chose the optimal solution during design and even operation of the information system.

In this paper we propose a common approach (Birta & Arbez, 2007) based on modelling and simulation. The aim of simulation is to calculate some performance metrics which should allow to compare different configuration taking into consideration technical (like performance) and economical (like price) aspects.

There is a large number of event driven computer network simulators, like OPNET, NS-2, QualNet, OMNeT++ or SSFNet/PRIME SSF(Liu, 2006; Nicol et al., 2003). However, they are mainly focused on a low level simulation (TCP/IP packets).

It is obvious that increasing the system details causes the simulation becoming useless due to the computational complexity and a large number of required parameter values to be given. On the other hand a high level of modelling could not allow to record required data for system measure calculation. Therefore, the level of system model details should be defined by requirements of the system measure calculation (Walkowiak, 2009).

Modelling and simulation based on TCP/IP packets level results in a large number of events during simulation and therefore in a long simulation time. It is a very good approach if one plans to analyze the influence of the traffic on the network performance. However in modern information systems high speed local networks are used. In a result for a large number of information systems (except media streaming ones) the local network traffic influence on the whole system performance is negligible.

Therefore, we want to propose a novel approach based on a higher level then TCP/IP packets. We will focus on a business service realized by an information system (Gold et al., 2007) and functional aspects of the system, i.e. performance aspects of business service realized by an information system (like buying a book in the internet bookstore). We assume that the main goal, taken into consideration during design and operation of the CIS, is to fulfil the user requirements, which could be seen as some requirements to perform a user tasks within a given time limit. Therefore, the presented in the chapter modelling and simulation will be focused on a process of execution of a user request, understand as a sequence of task realised on technical services provided by the system.

The structure of the chapter is as follows. In Section 2, a model of information system is given. In Section 3, information on simulator implementation is given, next exemplars information system is analysed and simulation results are presented. It is followed by information on graphical user interface. Finally, there are conclusions and plans for further work.

2. Computer information system modelling

As it was mentioned in the introduction we decided to analyze the CIS from the business service point of view. Generally speaking users of the system are generating tasks which are being realized by the CIS. The task to be realized requires some services presented in the system. A realization of the system service needs a defined set of technical resources. Moreover, the services has to be allocated on a given host. Therefore, we can model CIS as a 4-tuple (Walkowiak, 2009):

$$CIS = \langle Client, BS, TI, Conf \rangle \quad (1)$$

- Client* - finite set of clients,
BS - business service, a finite set of service components,
TI - technical infrastructure,
Conf - information system configuration.

During modelling of the technical infrastructure we have to take into consideration functional aspects of CIS. Therefore, the technical infrastructure of the computer system could be modelled as a pair:

$$TI = \langle H, N \rangle \quad (2)$$

where: *H* - set of hosts (computers); *N* - computer network.

We have assumed that the aspects of TCP/IP traffic are negligible therefore we will model the network communication as a random delay. Therefore, the *N* is a function which gives a value of time of sending a packet form one host (v_i) to another (v_j). The time delay is modelled by a Gaussian distribution with a standard deviation equal to 10% of mean value.

The main technical infrastructure of the CIS are hosts. Each host is described by its functional parameters:

- server name (unique in the system),
- host performance parameter - the real value which is a base for calculating the task processing time (described later),
- set of technical services (i.e. apache web server, tomcat, MySQL database), each technical service is described by a name and a limit of tasks concurrently being executed.

We have distinguished a special kind of technical service witch models a load balancer (Aweya et al., 2002). A load balancer is described by its name and a limit of tasks (like all technical services) and additionally by a list of technical services, it sends requests to.

The *BS* is a set of services based on business logic, that can be loaded and repeatedly used for concrete business handling process (i.e. ticketing service, banking, VoIP, etc). Business service can be seen as a set of service components and tasks, that are used to provide service in accordance with business logic for this process (Michalska & Walkowiak, 2008).

Therefore, *BS* is modelled as a set of business service components (*BSC*), (i.e. authentication, data base service, web service, etc.), where each business service component is described a name, reference to a technical service and host describing allocation of business service component on the technical infrastructure and a set of tasks. Tasks are the lowest level observable entities in the modelled system. It can be seen as a request and response form one service component to another. We have distinguished two kinds of task: local and external. If request is send to service component and this component is able to respond without asking other service component than this tasks is assumed to be local. If request is send to service component and this component must ask another service component for response then than this tasks is assumed to be external. Each task is described by its name, task processing time parameter and in case of external task by a sequence of task calls. Each task call is defined by a name of business service component and task name within this business service component and time-out parameter.

System configuration (*Conf*) is a function that gives the assignments of each service components to a technical service and therefore to hosts since a technical set is placed on a given host. In case of service component assigned in a configuration to a load balancing technical service the tasks included in a given service component are being realised on one of technical services (and therefore hosts) defined in the load balancer configuration.

The client model (*Client*) consist of set of users where each user is defined by its allocation (host name), replicate parameter (number of concurrently ruing users of given type), set of activities (name and a sequence of task calls) and inter-activity delay time (modelled by a Gaussian distribution).

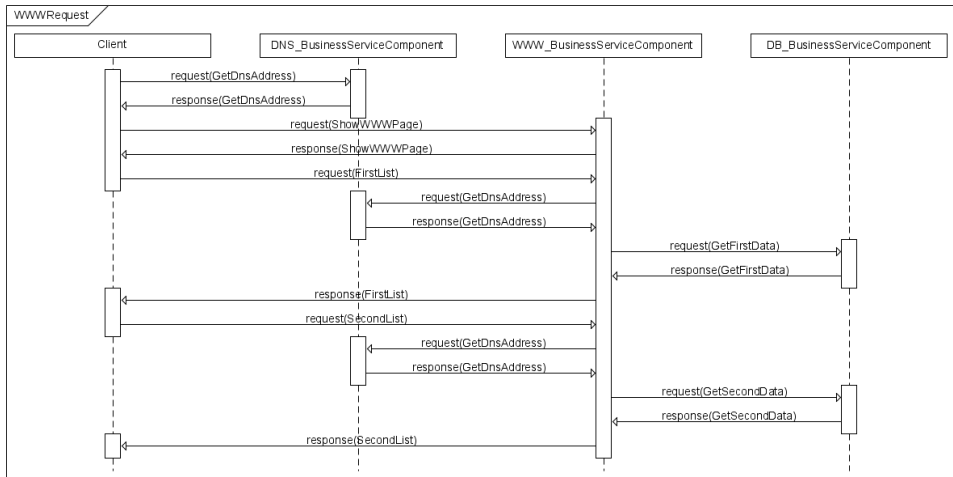


Fig. 1. Task and business services interaction

Summarising, a user initiate the communication requesting some tasks on a host, it could require a request to another host or hosts, after the task execution hosts responds to requesting server, and finally the user receives the respond. Requests and responds of each task gives a sequence of a user task execution as presented on exemplar Fig. 1.

The user request execution time in the system is calculated as a sum of times required for TCP/IP communication and times of tasks processing on a given host.

The request is understood as correctly answered if answers for each requests in a sequence of a user task execution were given within defined time limit (time-out parameter of each request in *BS* model) and if a number of tasks executed on a given technical service is not exceeding the limit parameter (parameter of *TI* model).

The user request execution time in the system is calculated as a sum of times required for TCP/IP communication (modelled by a random value) and times of tasks processing on a given host. The task processing time is equal to the task processing time parameter multiplied by a number of other task processed on the same host in the same time and divided by a the host performance parameter. Since the number of tasks is changing in simulation time, the processing time is updated each time a task finish the execution or a new task is starting to be processed.

Let $\tau_1, \tau_2, \dots, \tau_e$ be a time moments when a task (t_j^i) with some execution time ($executiontime(t_j^i)$) is starting or finishing processing on a host $h = allocation(t_j^i)$. Let $number(h, \tau)$ denotes a number of task being processed at time τ on host h . It is not taking into account tasks which requests tasks on other hosts and waits for responses. Therefore, the time when task t_j^i finishes its execution τ_e has to fulfill a following rule:

$$\sum_{k=2}^e (\tau_k - \tau_{k-1}) \frac{performance(h)}{number(h, \tau_{k-1})} = executiontime(t_j^i) \quad (3)$$

Having above notation the task processing time is equal to:

$$pt(t_j^i) = \tau_e - \tau_1 \quad (4)$$

3. Task-level simulator

Once a model has been developed, it is executed on a computer. It is done by a computer program which steps through time. One way of doing it is so called event-simulation. Which is based on a idea of event, which could is described by time of event occurring, type of event (in case of CIS it could be host failure) and element or set of elements of the system on which event has its influence. The simulation is done by analyzing a queue of event (sorted by time of event occurring) while updating the states of system elements according to rules related to a proper type of event.

As it was described in section 2, the network connections are modelled as a random delays. Therefore, we were not able to use mentioned in the introduction computer network simulators but we have to develop a new one (Walkowiak, 2009). The event-simulation

program could be written in general purpose programming language (like C++), in fast prototyping environment (like Matlab) or special purpose discrete-event simulation kernels. One of such kernels, is the Scalable Simulation Framework (SSF) (Nicol et al., 2003) which is used for SSFNet (Nicol et al., 2003) computer network simulator. SSF is an object-oriented API - a collection of class interfaces with prototype implementations. It is available in C++ and Java. SSF API defines just five base classes: Entity, inChannel, outChannel, Process, and Event. The communication between entities and delivery of events is done by channels (channel mappings connects entities).

For the purpose of simulating CIS we have used Parallel Real-time Immersive Modeling Environment (PRIME) (Liu, 2006) implementation of SSF due to much better documentation than available for original SSF. We have developed a generic class (named BSOBJECT) derived from SSF Entity which is a base of classes modeling CIS objects: host and client which models the behavior of CIS presented in section 2. Each object of BSOBJECT class is connected with all other objects of that type by SSF channels what allows communication between them. In the first approach we have realized each client as a separated object. However, in case of increasing of the number of replicated clients the number of channels increases in power of two resulting in a large memory consumption and a long time for initialization simulation objects. Therefore, we have changed the implementation, and each replicated client is represented by one object.

The developed simulator is called SSF.BS (from SSF - the simulation framework and BS - business service).

4. Computer information system simulation analysis

4.1 First case study

For testing purposes of presented CIS system model (section 2) and developed extension of SSF (SSF.BS, section 3) we have analysed a case study information system. It consists of one type of client placed somewhere in internet, firewall, three hosts (Figure 2), three technical services and three business service components. An interaction between a client and tasks of each business service component is presented on UML diagram in Figure 1. The CIS structure as well as other functional parameters were described in a DML file (see example in Figure 3). The Domain Modeling Language (DML) (Nicol et al., 2003) is a SSF specific text-based language which includes a hierarchical list of attributes used to describe the topology of the model and model attributes values.

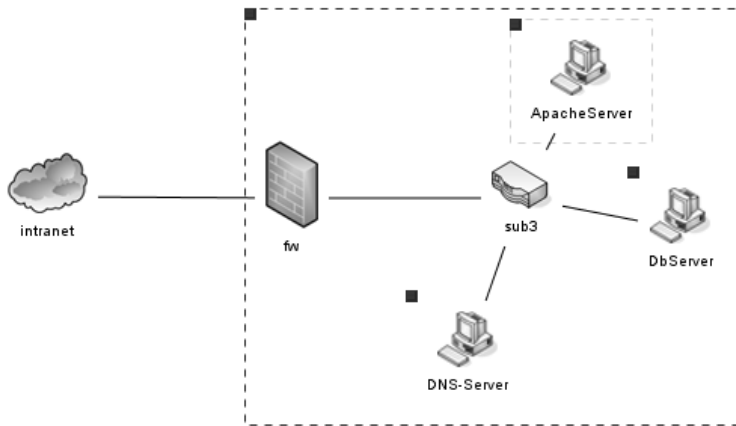


Fig. 2. Case study system overview

```

Net [
Host [
  Name DNS-Server
  Service [
    Name DNSService
    Limit 110
    LocalTask [
      Name GetDnsAddress
      Time 0.01]]]
  ...
Client [
  Name Client
  Replicate 1000
  Sleep 10.0
  Activity [
    Name WWWRequest
    TaskCall [
      Host DNS-Server
      Service DNSService
      Task GetDnsAddress
    ]
  ]
  ...
]

```

Fig. 3. Exemplar CIS description in DML file

In the presented information system we have observed the response time to a client request in a function of number of clients. The achieved results are presented in Figure 4.

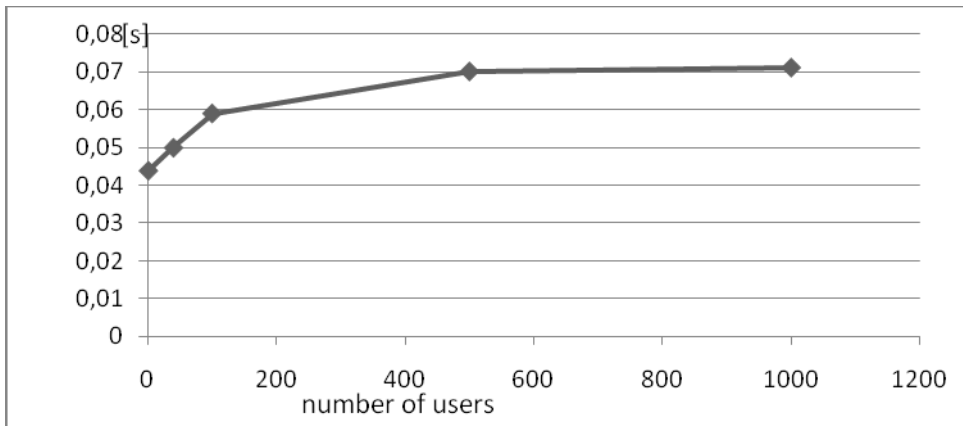


Fig. 4. Response time to users requests in a function of number of concurrent users

4.2. Simulator performance analysis

Next, we have tested the SSF.BS simulator performance and scalability. We calculated the time of running one batch of simulation of the exemplar IS described in previous chapter on a 2.80 GHz Intel Core Duo machine. We have compared the performance results with PWR.SSF.Net simulator (Zyla & Caban 2008) developed in Java. The CIS model used in PWR.SSF.Net differs from SSF.BS mainly in a method of calculation a task performance time and therefore the results of simulating cannot be compared. As it could be noticed on Figure 5 & 6 the presented in the paper simulator (SSF.BS) simulates the CIS in shorter time, and a difference with PWR.SSF.Net is increasing with an increase of number of users.

For a number of concurrent users less than 300 (Figure 5) the SSF.BS is 10 times faster than PWR.SSF.Net. The main reason of this difference is the level of modelling details. In both cases simulators perform similar number of events per second. However, PWR.SSF.Net simulates the transmission of TCP/IP packets whereas SSF.BS works on higher level the tasks and therefore in case of presented here approach the number of events is smaller.

Not, only computational complexity of SSF.BS is lower than PWR.SSF.Net but also the usage of memory for SSF.BS is much smaller. For a case study example the SSF.BS requires 1.8 Mbytes for 0.1 client requests per second upto 4.8 Mbytes for a 1000 concurrent users. In case of PWR.SSF.Net it is hard to state the memory usage due to the memory management techniques in Java. This is the problem of enlarging the difference of speed between analysed simulators. For number of clients more then 300 (Figure 6) Java based PWR.SSF.Net starts to have problems with memory management and large number of processing time is used by JVM garbage collector (even Java based simulator was started 1 Gbyte memory limit). It results in 1000 faster simulation of SSF.BS in case of 1000 concurrent users.

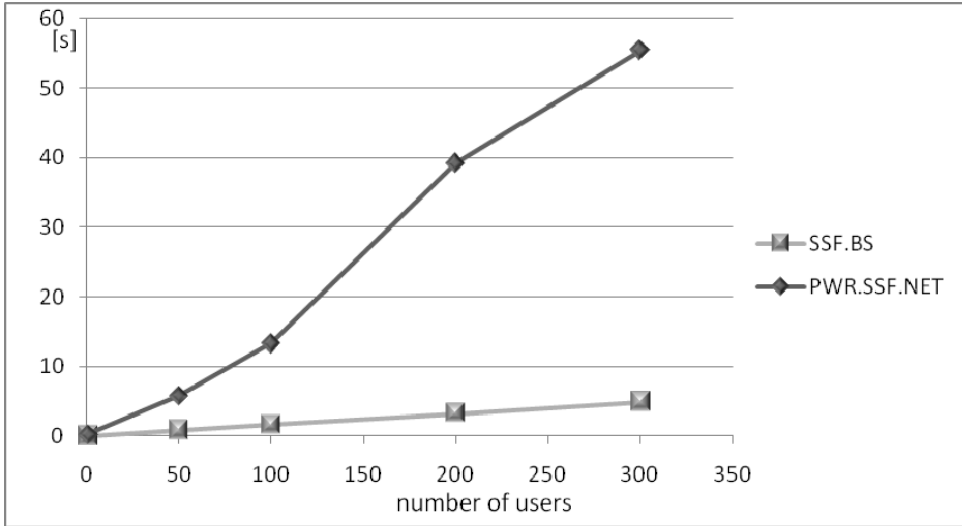


Fig. 5. Simulation time (time of running the simulator) for case study system in a function of number of users (till 300 concurrent users)

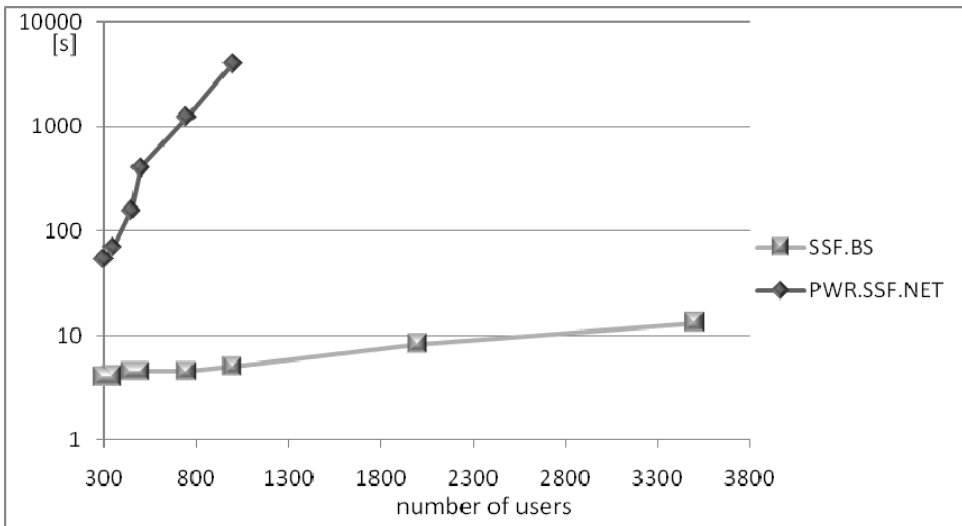


Fig. 6. Simulation time (time of running the simulator) for case study system in a function of number of users (for more than 300 users)

4.3. Second case study – load balancer

A very common technique of achieving height availability of their services in CIS is using a load balancer. Load balancer allows a traffic distribution among replicated services on a server farm. Therefore, the most common load balancing algorithm - *round robin* (Aweya, et al. 2002) - has been implemented in the SSF.BS.

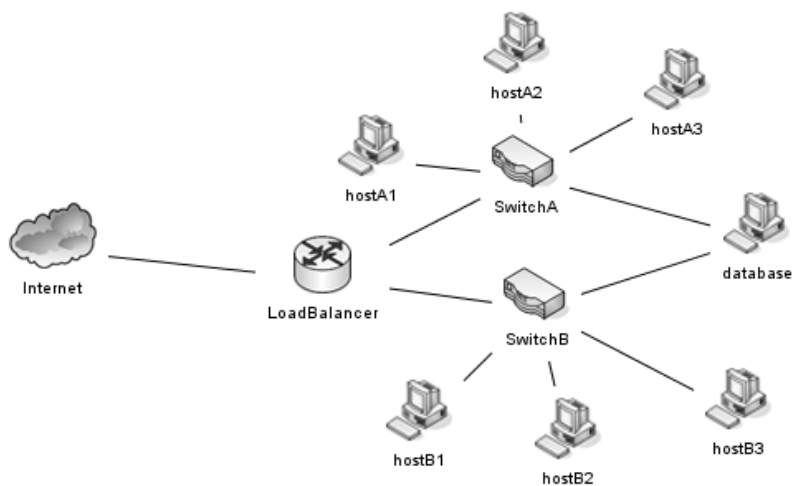


Fig. 7. Load balancer case study system overview

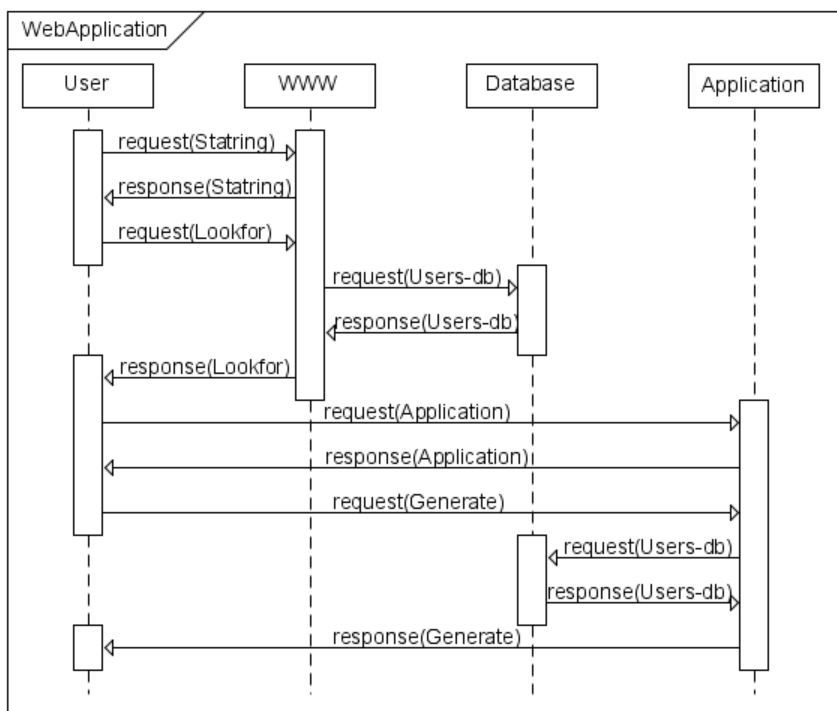


Fig. 8. Task and business services interaction for case study

For the case study analysis of CIS with load balancing we propose an exemplar service system illustrated in Fig.7 . Essentially the test-bed system consists of two server farms A (included host „hostA1“-„hostA3“) and B (included host „hostB1“-„hostB3“) and a database server. Both farms are connected with LoadBalancer as a gate to internet users. For the case study, let us imagine, that this system is responsible for some Web Application that allows searching the database and executes a Tomcat based application. Fig. 8 shows choreography of this service, based on three service components. WWW service component has been replicated on hosts: A1-A3, Application of on hosts: B1- B3 and Database is not replicated is placed on one host. For this scenario two configuration has been proposed: first (I) standard and second (II) with all hosts with doubled performance parameter.

The achieved simulation results, the response time to user requests in a function of number of concurrent users is presented in Figure 9. The simulation time was set to 1000 seconds. The limit of concurrent tasks for all technical services was equal to 1000, whereas the interactivity delay time equal to 1 s. As it could be expected the response time for configurations II is almost twice shorter than for configuration I. However, if we slightly change configuration II, setting the performance of database host equal to the value used in configurations I the resulting response time will be very similar to results of configuration I. These small experiment shows the ability of simulator to compare performance of different system configurations.

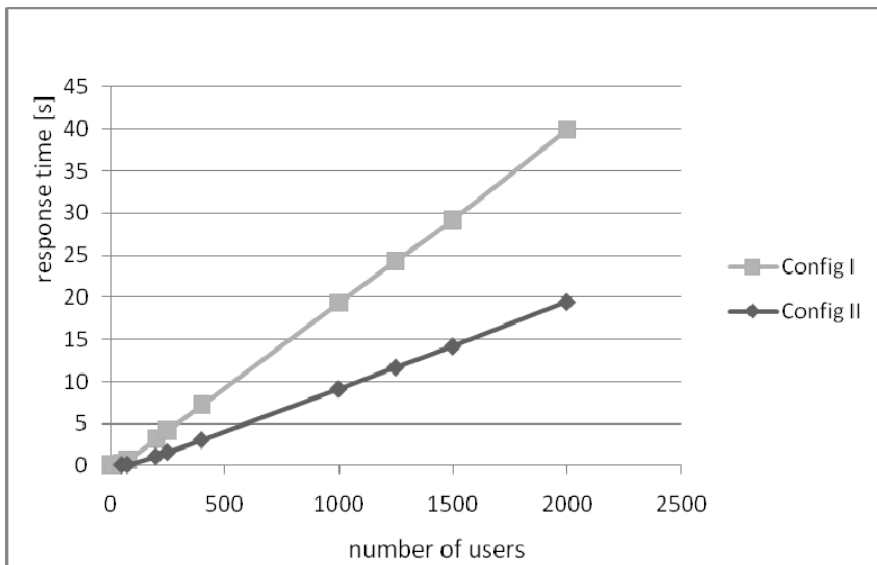


Fig. 9. Response time to users requests in a function of number of concurrent users for two configurations of load balancer case study

5. Graphical interface

The previous section showed the possibilities of using SSF.BS simulator and its good computational performance capabilities. However, nowadays the practical usage of any computer tool requires a good graphical interface. As it was mentioned in the section 3, all

input information of modelled CIS is described in DML text file. Even the DML file format is simple (Figure 3), it is difficult for a human being to describe a CIS with large number of host and sophisticated service interaction without any error in text file.

Within the framework of DESEREC EU grant (<http://www.deserec.eu>) a Java based graphical tool called "Integrated Analysis Environment" (IAE) was developed (Michalska & Walkowiak, 2008b) for a usage of PWR.SSF.Net simulator. After a few changes in IAE it was adopted to SSF.BS simulator.

In IAE we took into consideration an inconvenient format of Domain Modelling Language and we proposed its XML representation with all supplements attributes of proposed extended simulation framework - called XXML. Creation of XXML language gave many processing possibilities. IAE framework using JAXB techniques and implemented translation methods creates one model (XXML) from other modelling languages: system infrastructure from SDL (System Description Language, <http://www.positif.org/>) and task interaction from WS-CDL (WebServices Choreography Description Language, <http://www.w3.org/>). This XXML model is visualized showing the structure of the network and it's element (Figure 10). Each network element has several functional parameters and user can graphically edit this information. In proposed framework user is able to put its own variables and attributes based on XXML specification or use extend models (i.e. consumption model, operational configuration model) to simplified its work.

After setting up all parameters of network elements and service components the user is able to perform simulation. It is done by transforming XXML into DML. The resulting DML file is then simulated. Simulation is integrated into IAE since both tools are developed in Java therefore user can see on the screen text output from the simulator on-line. The results from simulation (output file from simulator) are caught by IAE and response time to user requests is calculated and displayed.

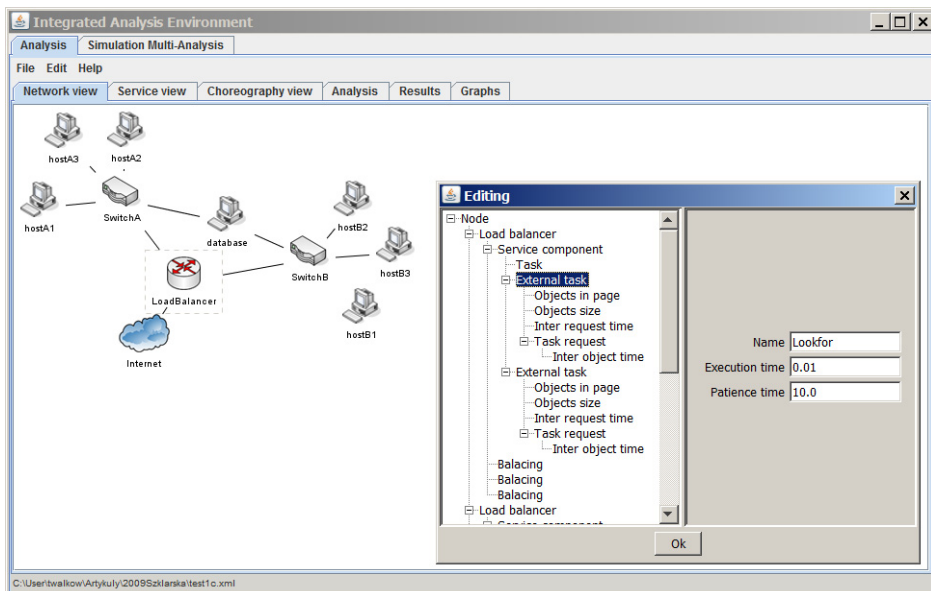


Fig. 10. Integrated Analysis Environment - screenshot

6. Conclusion

We have presented a simulation approach to functional analysis of complex information systems. Developed simulation software allows to analyze the effectiveness (understood in given exemplar as the response time to a client request) of a given configuration of computer system. Changes in a host performance or in a number of clients can be easily verified. Also, some economic analysis could be done following the idea presented in (Walkowiak & Mazurkiewicz, 2005). The implementation of CIS simulator done based on SSF allows to apply in a simple and fast way changes in the CIS model. Also the time performance of SSF kernel results in a very effective simulator of CIS.

We are now working on implementing other load balancing algorithms what should allow to analyze a wider range of enterprise information systems and compare different load balancing algorithms.

We also plan to extend the model and simulator to include the reliability model of technical infrastructure components. It should allow to measure the availability of a business service in a function of functional and reliability parameters of information systems components.

7. References

- Avižienis, A.; Laprie, J.; Randell, B. (2000). Fundamental Concepts of Dependability. *Proceedings of 3rd Information Survivability Workshop (ISW-2000)*, Boston, Massachusetts
- Aweya, J.; Ouellette, M.; Montuno, D.; Doray, B.; Felske, K. (2002). An adaptive load balancing scheme for web servers. *International Journal of Network Management*, Vol. 12
- Birta, L.; Arbez, G. (2007). *Modelling and Simulation: Exploring Dynamic System Behaviour*. Springer, London
- Gold, N.; Knight, C.; Mohan, A.; Munro, M. (2004). Understanding service-oriented software. *IEEE Software*, Vol. 21, 71–77
- Liu, J. (2006). Parallel Real-time Immersive Modeling Environment (PRIME), Scalable Simulation Framework (SSF), User's manual. Colorado School of Mines Department of Mathematical and Computer Sciences, 2006, [Online]. Available: <http://prime.mines.edu/>
- Nicol, D.; Liu, J.; Liljenstam, M.; Guanhua, Y. (2003). Simulation of large scale networks using SSF. *Proceedings of the 2003 Winter Simulation Conference*, Vol. 1, pp. 650–657, New Orleans,
- Michalska, K.; Walkowiak, T. (2008). Hierarchical Approach to Dependability Analysis of Information Systems by Modeling and Simulation. *Proceedings of the 2008 Second international Conference on Emerging Security information, Systems and Technologies*, , pp. 356-361 Cap Esterel, IEEE Computer Society, Washington
- Walkowiak, T.; Mazurkiewicz, J. (2005) Reliability and Functional Analysis of Discrete Transport System with Dispatcher. *Advances in Safety and Reliability, European Safety and Reliability Conference – ESREL 2005*, Gdynia, pp. 2017-2023, Taylor & Francis Group, London
- Walkowiak, T. (2009). Information systems performance analysis using task-level simulator, *Proceedings of International Conference on Dependability of Computer Systems*, pp. 218–225, Brunow, IEEE Computer Society Press, Los Alamitos
- Zyla, M.; Caban, D. (2008). Dependability Analysis of SOA systems. *Proceedings of International Conference on Dependability of Computer Systems*, pp. 301–306, Szklarska Poreba, IEEE Computer Society Press, Los Alamitos

Modelling equipment deterioration vs. maintenance policy in dependability analysis

Jarosław Sugier
Wrocław University of Technology
Poland

George J. Anders
Technical University of Łódź
Poland

1. Introduction

Effective and efficient maintenance is a significant factor in operation of today's complex computer systems. Selecting the optimal maintenance strategy must take numerous issues into account and among them reliability and economic factors are often of equal importance. On one side, it is obvious that for successful system operation failures must be avoided and this opts for extensive and frequent maintenance activities. On the other, superfluous maintenance may result in very large and unnecessary cost. Finding a reasonable balance between these two is a key point in efficient system operation.

This text describes Asset Risk Manager (ARM) – a computer software package provided as a decision support tool for a person selecting maintenance activities. Its main task is to help in evaluation of risks and costs associated with choosing different maintenance strategies. Rather than searching for a solution to a problem: “what maintenance strategy would lead to the best dependability parameters of system operation”, in our approach different maintenance scenarios can be examined in “what-if” studies and their reliability and economic effects can be estimated.

The main idea of the approach is based on the concept of a life curve and discounted cost used to study the effect of equipment ageing under different maintenance policies. First, the deterioration process in the presence of maintenance activities is described by a Markov model and then its various characteristics are used to develop the equipment life curve and to quantify other reliability parameters. Based on these data, effects of various “what-if” maintenance scenarios can be visualized and their efficiency compared. Simple life curves computed from the model can be combined to represent equipment deterioration undergoing diverse maintenance actions, while computing other parameters of the model allows evaluating additional factors, such as probability of equipment failure.

Special care is paid to one particular problem: having a model that describes the deterioration of an element that undergoes some maintenance policy with particular repair frequencies, it is often needed to create a model representing the same element being subjected to a new policy that differs only in repair frequencies. The method proposed for

creation of such a model adjusts the initial one through fine-tuning probabilities of the repair states in an iterative process that converges to the desired goal. Discussion of different possible approximation methods applied during the adjustment is included and effectiveness of this approach is illustrated with practical examples.

The ARM system itself has been initially presented in (Anders & Sugier, 2006). This text extends that presentation with additional discussion of the method for Markov model adjustment and its impact on new results that can be included in the studies (Sugier & Anders, 2007).

2. Modelling the ageing process in the presence of maintenance activities

In the proposed approach it is assumed that the equipment will deteriorate in time and, if not maintained, will eventually fail. If the deterioration process is discovered, preventive maintenance is performed which can often restore the condition of the equipment. Such a maintenance activity will return the system to a specific state of deterioration, whereas repair after failure will restore to “as new” condition (Hughes & Russell, 2005; Anders & Endrenyi, 2004).

Markov models, which form the underlying structure of the models investigated here, have been applied during planning and operation of large networks (IEEE/PES Task Force, 2001). Equipment aging processes with non-exponential time of sojourn in the states can be represented by several series of stages (Li & Guo, 2006). Each stage can be represented as a state in the Markov process so that the non-Markovian processes can be transformed into Markovian processes (IEEE/PES Task Force, 2001; Singh & Billinton, 1997; Tomasevicz & Asgarpour, to be published). Fuzzy Markov models have also been developed in which uncertainties in transition rates / probabilities are represented by fuzzy values (Mohanta et al., 2005; Duque & Morinigo, 2004; Cugnasca et al., 1999; Ge et al., 2007). In these models, fuzzy arithmetic was applied to mimic the crisp Markov process calculations which are computationally tedious and even more so when the number of states increases.

2.1 The life curves

A convenient way to represent the deterioration process is by a *life curve* of the equipment (Anders & Endrenyi, 2004). Such a curve shows the relationship between asset condition, expressed in either engineering or financial terms, and time. Since there are many uncertainties related to the prediction of equipment life, probabilistic analysis must be applied to construct and evaluate life curves. Fig. 1 (a) shows an example of a simple life curve of some equipment that models its continuous deterioration up to the point of failure. Fig. 1 (b) illustrates application of this curve in a case study of some specific scenario in which equipment refurbishment and equipment failure occur.

2.2 The ageing process

There are three major factors that contribute to the ageing behaviour of equipment: physical characteristics, operating practices, and the maintenance policy. Of these three aspects the last one relates to events and actions that should be properly incorporated in the model. The maintenance policy components that must be recognized in the model are: monitoring or inspection (how is the equipment state determined), the decision process (what

determines the outcome of the decision), and finally, the maintenance actions (or possible decision outcomes).

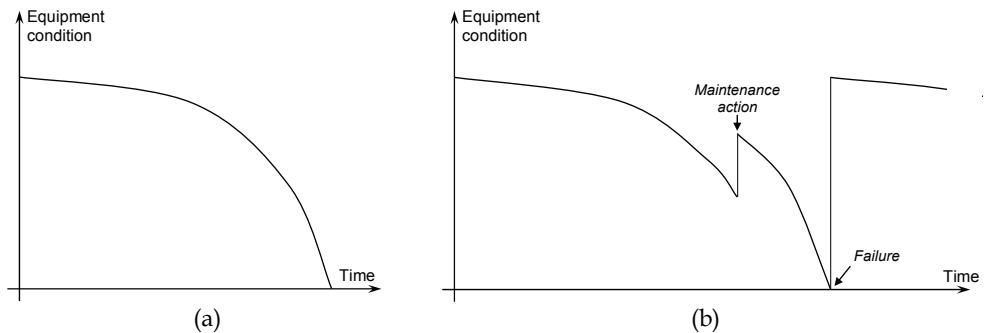


Fig. 1. Life curve of an equipment (a) and its application to modelling equipment condition over some time period (b).

In practical circumstances, an important requirement for the determination of the remaining life of the equipment is the establishing its current state of deterioration. Even though at the present state of development no perfect diagnostic test exists, monitoring and testing techniques may permit approximate quantitative evaluation of the state of the system. It is assumed that four deterioration states can be identified with reasonable accuracy: (a) normal state, (b) minor deterioration, (c) significant (or major) deterioration, and (d) equipment failure. Furthermore, the state identification is accomplished through the use of scheduled inspections. Decision events generally correspond to inspection events, but can be triggered by observations acquired through continuous monitoring. The decision process will be affected by what state the equipment is in, and also by external factors such as economics, current load level of the equipment, its anticipated load level and so on.

2.3 The model

All of the above assumptions about the ageing process and maintenance activities can be incorporated in an appropriate state-space (Markov) model. It consists of the states the equipment can assume in the process, and the possible transitions between them. In a Markov model the rates associated with the transitions are assumed to be constant in time. The development described in this paper uses model of Asset Maintenance Planner (Anders & Maciejewski, 2006; Anders & Leite da Silva, 2000). The AMP model is designed for equipment exposed to deterioration but undergoing maintenance at prescribed times. It computes the probabilities, frequencies and mean durations of the states of such equipment. The basic ideas in the AMP model are the probabilistic representation of the deterioration process through discrete stages, and the provision of a link between deterioration and maintenance.

For structure of a typical AMP model see Fig. 2. In most situations, it is sufficient to represent deterioration by three stages: an initial (D1), a minor (D2), and a major (D3) stage. This last is followed, in due time, by equipment failure (F) which requires extensive repair or replacement.

In order to slow deterioration and thereby extend equipment lifetime, the operator will carry out maintenance according to some pre-defined policy. In the model of Fig. 2, regular inspections (Is) are performed which result in decisions to continue with minor (Ms1) or major (Ms2) maintenance or do nothing (with the state number $s = 1, 2$ or 3). The expected result of all maintenance activities is a single-step improvement in the deterioration chain; however, allowances are made for cases where no improvement is achieved or even where some damage is done through human error in carrying out the maintenance resulting in the next stage of deterioration.

The choice probabilities (at the points of decision making) and the probabilities associated with the various possible outcomes are based on user input and can be estimated e.g. from historical records or operator expertise. For the needs of further tuning of the model the probabilities linked to transitions to the maintenance states M_{si} are the most important ones as they are directly related to the repair frequencies. These probabilities will be denoted as P^{sr} ($P^{11}, P^{12}, \dots, P^{32}$), where $s =$ state number and $r =$ repair index.

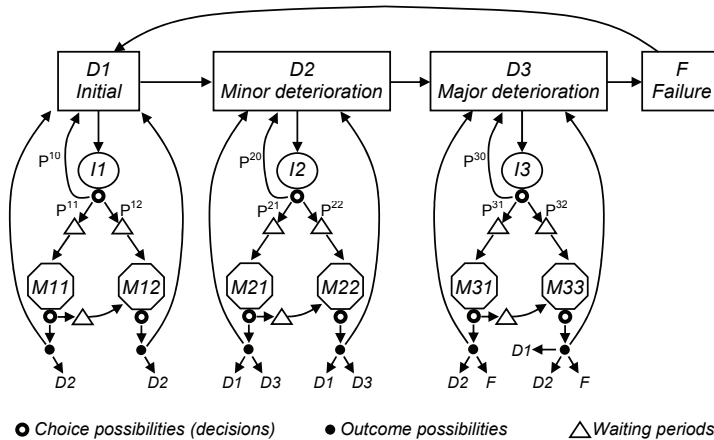


Fig. 2. Model of the ageing process for equipment undergoing inspections and maintenance activities. Decision probabilities after inspection states are placed by respective transitions. $K = 3$, $R = 2$.

Mathematically, the model in Fig. 2 can be represented by a Markov process, and solved by well-known procedures. The solution will yield all the state probabilities, frequencies and mean durations. Another technique, employed for computing the so-called first passage times (FPT) between states, will provide the average times for first reaching any state from any other state. If the end-state is F, the FPT's are the mean remaining lifetimes from any of the initiating states.

3. Adjusting model parameters

Preparing the Markov model for some specific equipment is not an easy task and requires participation of an expert. The goal is to create the model representing closely real-life deterioration process known from the records that usually describe average equipment

operation under regular maintenance policy with some specific frequencies of inspections and repairs. Compliance with these frequencies in behaviour of the model is a very desirable feature that verifies its trustworthiness.

This section describes a method of model adjustment that aims at reaching such a compliance (Sugier & Anders, 2007). It can be used also for a different task: fully automatic generation of a model for a new maintenance policy with modified frequencies of repairs.

3.1 The method

Let K represents number of deterioration states and R - number of repairs in the model under consideration. Also, let P^{sr} = probability of selecting maintenance r in state s (assigned to decision after state I_s) and P^{s0} = probability of returning to state D_s from inspection I_s (situation when no maintenance is scheduled as a result of the inspection). Then for all states $s = 1 \dots K$:

$$P^{s0} + \sum_r P^{sr} = 1 \quad (1)$$

Let F^r represents frequency of repair r acquired through solving the model. The problem of model tuning can be formulated as follows:

Given an initial Markov model M_0 , constructed as above and producing frequencies of repairs $\mathbf{F}_0 = [F_0^0, F_0^1, \dots, F_0^R]$, adjust probabilities P^{sr} so that some goal frequencies \mathbf{F}_G are achieved.

Typically, the vector \mathbf{F}_G represents observed historical values of the frequencies of various repairs. In the proposed solution, a sequence of tuned models $M_0, M_1, M_2, \dots, M_N$ is evaluated with each consecutive model approximating desired goal with a better accuracy. The tuning procedure begins with an initial model M_0 and then in each iteration the following steps are performed:

- 1° For the current model M_i compute vector of repair frequencies \mathbf{F}_i .
- 2° Evaluate an error of M_i as a distance between vectors \mathbf{F}_G and \mathbf{F}_i .
- 3° If the error is within the user-defined limit consider M_i as the final tuned model and stop the procedure ($N = i$); otherwise proceed to the next step.
- 4° Create model M_{i+1} through tuning values of P_i^{sr} ; adjust also P_i^{s0} according to (1).
- 5° Go to step 1° and proceed with the next iteration.

The error computed in step 2° can be expressed in many ways. As the frequencies of repairs may vary in a broad range within one vector \mathbf{F}_i , yet values of all are significant in model interpretation, the relative measures work best in practice:

$$\|\mathbf{F}_G - \mathbf{F}_i\| = \frac{1}{R} \sum_{r=1}^R |F_i^r / F_G^r - 1|$$

or

$$\|\mathbf{F}_G - \mathbf{F}_i\| = \max_r |F_i^r / F_G^r - 1|. \quad (2)$$

The latter formula is more restrictive and was used in examples of this work.

3.2 Approximation of model probabilities

Of all the steps outlined in the previous section, it is clear that adjusting probabilities P_i^{sr} in step 4° is the heart of the whole procedure.

In general, the probabilities represent $K \cdot R$ free parameters and their uncontrolled modification could lead to serious deformation of the model. To avoid this, a restrictive assumption is made: if the probability of some particular maintenance must be altered, it is modified proportionally in all deterioration states, so that at all times

$$P_0^{1r} : P_0^{2r} : \dots : P_0^{Kr} \sim P_i^{1r} : P_i^{2r} : \dots : P_i^{Kr} . \quad (3)$$

for all repairs ($r = 1 \dots R$).

This assumption also significantly reduces dimensionality of the problem, as now only a vector of R scaling factors $\mathbf{X}_{i+1} = [X_{i+1}^1, X_{i+1}^2, \dots, X_{i+1}^R]$ must be found to compute probabilities of the next model M_{i+1} :

$$P_{i+1}^{sr} = X_{i+1}^r \cdot P_0^{sr}, \quad r = 1 \dots R, \quad s = 1 \dots K$$

Moreover, although frequency of a repair r depends actually on probabilities of all repairs (modifying probability of one repair changes, among others, state durations in the whole model, thus it changes frequencies of all states) it can be assumed that in case of a single-step small adjustment its dependence on repairs other than r can be neglected and

$$F_i^r(X_i^1, X_i^2, \dots, X_i^R) \approx F_i^r(X_i^r). \quad (4)$$

With these assumptions generation of a new model in step 4° is reduced to finding roots of R non-linear equations in the form of $F_i^r(X_i^r) = F_G^r$. This can be accomplished with one of standard numerical algorithms.

For the needs of development described in this work the following three approximation algorithms has been implemented and verified on practical examples: (A) Newton method working on linear approximation of $F_i^r(0)$, (B) the secant method and (C) the false position (*falsi*) method.

(A) Newton method On Linear Approximation (NOLA)

In this solution it is assumed that $F_i^r(0)$ is a linear function defined by points $F_i^r(X_i)$ (computed for the current model M_i in step 1°) and $F_i^r(0)$ (which is zero). Then the scaling factor for any repair r is taken simply as:

$$X_{i+1}^r = F_G^r / F_i^r .$$

Applying these factors to all repair probabilities creates the next model M_{i+1} .

This method is very simple and may seem primitive but its noteworthy advantage lies in the fact that no other point than the current frequency $F_i^r(X_i)$ is required to compute the next approximation. As errors of the previous iteration steps do not accumulate, convergence is good from the very first iteration.

(B) The secant method

In this standard technique the function is approximated by the secant defined by the last two approximations in points X_{i-1}^r , X_i^r so that a new one is computed as:

$$X_{i+1}^r = X_i^r - \frac{X_i^r - X_{i-1}^r}{F_i^r - F_{i-1}^r} (F_i^r - F_G^r). \quad (5)$$

After that X_{i-1}^r is discarded; X_{i+1}^r and X_i^r are considered in the next iteration.

To start the procedure two initial points are needed. In this method it is proposed to choose the initial frequency of the model M_0 ($X_0^r = 1$) as the first point, while the second one is computed as in NOLA method above, i.e. $X_1^r = F_G^r / F_0^r$. Starting with these two, the next solutions are computed as in (5).

(C) The false position (*falsi*) method

In this approach X_{i+1}^r is computed as in (5) but the difference lies in choosing points for the next iteration. While in (B) always X_{i-1}^r is dropped, now the new solution X_{i+1}^r is paired with that one of X_i^r , X_{i-1}^r which lies on the opposite side of the root. In this way when (5) is applied the solution is bracketed between X_i^r and X_{i-1}^r (which is the essence of *falsi* method). As in (B), to begin the iteration the two initial points are needed but now they must lie on both sides of the root, i.e.

$$(F_0^r - F_G^r) \cdot (F_1^r - F_G^r) < 0. \quad (6)$$

Choosing such points may pose some difficulty. To avoid multiple sampling, it is proposed to select $X_0^r = 1$ (as previously) and then to compute X_1^r like in NOLA method but with some "overshoot" that would guarantee condition (6):

$$X_1^r = (F_G^r / F_0^r)^\alpha. \quad (7)$$

The parameter $\alpha > 1$ limits the overshoot effect. The overshoot must be sufficient to ensure (6) but, on the other hand, should not produce too much of an error because that would deteriorate convergence process during initial steps and would add extra iterations. In practice values of $\alpha = 1.5 \div 2.5$ work well. Even if the initial value of X_1^r computed this way does not meet (6) for particular value of α then (7) can be re-applied with α increased, although it should be noted that each such correction requires solving a new M_1 model and in effect this is the extra computational cost almost equal to that of the whole iteration.

3.3 Comparison of the methods

We shall now discuss effectiveness of the above three approximation methods using a sample Markov model tuned for four different repair frequencies. The final result presented to the user - life curves that were obtained from the tuned models - is shown in Fig. 3.

The model of the equipment consisted of 3 deterioration states and 3 repairs ($K = R = 3$), with $Ms1$ representing minor, $Ms2$ medium and $Ms3$ major repair. The life curve estimated

from model M_0 is shown as case 1 and serves as a point of reference. Cases 2 to 5 were created through adjusting M_0 to modified maintenance as follows:

case 2: frequencies of all repairs were reduced by half, $F_G = \frac{1}{2} F_0$

case 3: all repairs but major (M_s3) were removed, $F_G = [0, 0, F_0^3]$

case 4: frequencies of all repairs were reduced to 25%, $F_G = \frac{1}{4} F_0$

case 5: all repairs were removed, $F_G = [0, 0, 0]$.

All the three approximation methods (NOLA, secant and *falsi*) converged properly to the same set of probabilities that gave desired goal frequencies. For an example, Fig. 4 compares convergence rate of the methods in the case 2, i.e. during adjustment towards repair frequencies decreased by 50%. The value of the relative error (2) was reduced from initial 100% to 1% after just 3 iterations proving the high effectiveness of the proposed model tweaking. This also shows that simplifications (3) and (4) from section 3.2 are justifiable and do not deteriorate the approximation process.

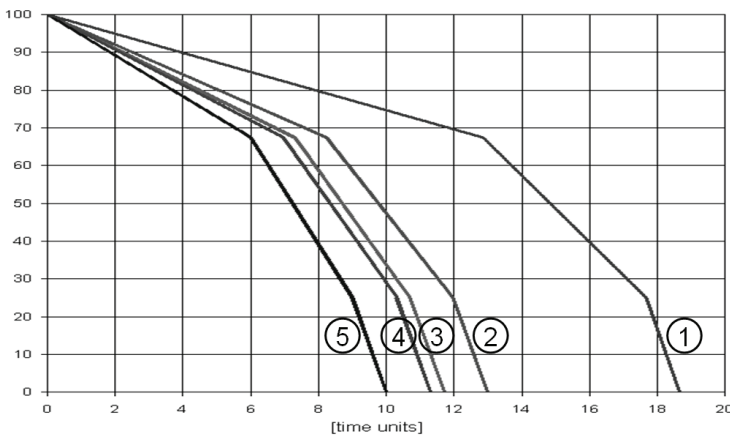


Fig. 3. Life curve of equipment for some default maintenance policy (1) and life curves generated from Markov models adjusted to modified policies (2-5).

What also can be seen for this specific case in Fig. 4 is that the three approximation methods, although significantly different from the mathematical point of view, yield very similar results during the first iterations 1 ÷ 3. The difference becomes visible starting from iteration no. 4 when, apparently, the secant method generated the approximating point that did not meet condition (6) and, effectively, lost this iteration reaching accuracy of the *falsi* method one step later. The same situation happened also in iterations no. 6 and 7. Compared to this, the NOLA method showed no such fluctuations and produced steady improvement in every step, although at a rate not as high as that of the *falsi* method. Fig. 5 presents the convergence rate in the other cases.

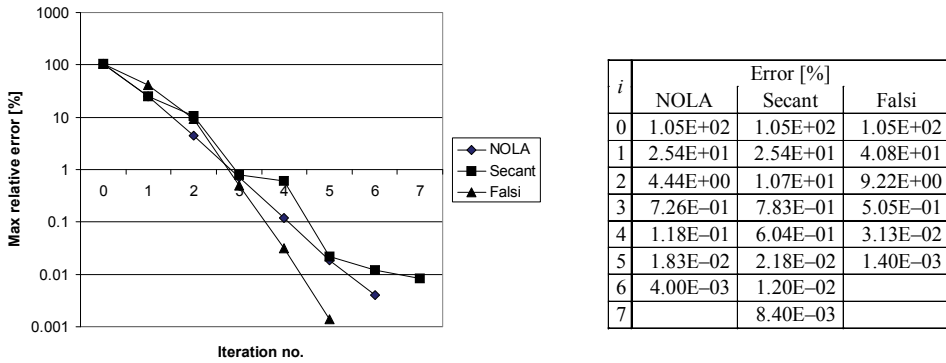


Fig. 4. Effectiveness of the three approximation methods in model tuning for $F_G = \frac{1}{2}F_0$.

Comparing the effectiveness of the methods it should be noted that although simplifications of the NOLA solution may seem critical, in practice it works quite well. As it was noted before, this method has one advantage over its more sophisticated rivals: since it does not depend on previous approximations, selection of the starting point is not so important and the accuracy during the first iterations is often better than in the secant or *falsi* methods. For example, in the case 2 (Fig. 4) NOLA method reached accuracy of 4.4% already after 2 iterations, while for secant and *falsi* methods the errors after two iterations were, respectively, 11% and 9.2%. Superiority of the latter methods, especially of the *falsi* algorithm, becomes undisputable in the later stages of approximation when the potential problems with initial selection of the starting points have been diminished.

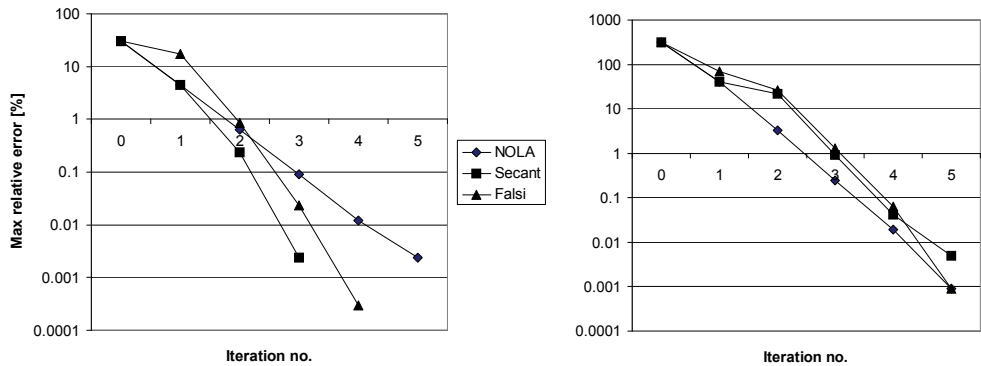


Fig. 5. Convergence rate in tuning for $F_G = [0, 0, F_0^3]$ (case 3, left) and $F_G = \frac{1}{4} F_0$ (case 4, right).

4. Asset Risk Manager

The Asset Risk Manager (ARM) is a software package which uses the concept of a life curve and discounted cost to study the effect of equipment ageing under different hypothetical maintenance strategies (Anders & Sugier, 2006). The curves generated by the program are based on Markov models that were presented in the two previous sections.

For the program to generate automatically the life curves, default Markov model for the equipment has to be built and stored in the computer database. This is done through the prior running of the AMP program by an expert user. Therefore, both AMP and ARM programs are closely related, and usually, should be run consecutively.

Implementation details of Markov models, tuning its parameters and all other internal particulars should not be visible to the non-expert end user. All final results are visualized either through an easy to comprehend idea of a life curve or through other well-known concepts of financial analysis. Still, prior to running the analysis some expert involvement is needed, largely in preparation, importing and adjusting AMP models.

4.1 User input

A typical study is described through a comprehensive set of parameters that are supplied by a non-expert end user. They fall into three broad categories.

(A) General data. The Markov model of the equipment in question and its current state of deterioration form the primary information that is the starting point to most of ARM computations. The Markov model represents the equipment with present maintenance policy and is selected from a database of imported AMP models which needs to be prepared by an expert in advance. Deterioration state, referred to as "Asset Condition" (AC) throughout the ARM, must be supplied by the end user as percentage of "as-new" condition. Besides, a number of additional general parameters need to be specified, such as the time horizon over which the analysis will be performed, discount and inflation rates for financial calculations etc.

(B) Description of the present maintenance policy. It is assumed that three types of maintenance repairs can be performed: minor, medium and major. These correspond to appropriate states in Markov model and not all of them must be actually present in the policy. For each repair user supplies its basic attributes, e.g. cost, duration and frequency.

(C) List of alternative actions. These are the hypothetical maintenance policies that decision-maker can choose from. Each action is defined as one of four types:

- continue as before (i.e. do not change the present policy),
- do nothing (i.e. stop all the repairs),
- refurbish,
- replace.

Apart from the first type, every action can be delayed for a defined amount of time. Additionally, for "non-empty" actions (i.e. any of the last two types) user must specify what to do in the period after action; the choices are:

- (a) to change type of equipment and / or
- (b) to change maintenance policy.

For every action user must also specify what to do in case of failure: whether to repair or replace failed equipment, its condition afterwards, cost of this operation etc. Thanks to these options a broad range of maintenance situations can be described and then analysed.

The first action on the list is always “Continue as before” and this is the base of reference for all the others. The ARM can be directed to compute life curves, cost curves, or probabilities of failure – for each action independently – and then to visualize computed data in many graphical forms to assist the decision-maker in effective action assessment.

It should be noted that while the need for some action (e.g., overhaul or change in maintenance policy) is identified at the present moment, the actual implementation will usually take place only after a certain delay during which the original maintenance policy is in effect. Using ARM it is possible to analyze effect of that delay on the cost and reliability parameters.

4.2 Life curves

As it has been pointed out before, computing the average first passage time (FPT) from the first deterioration state (D1) to the failure state (F) in the Markov model yields the average lifetime of the equipment, i.e. length of its life curve. On the other hand, solving the model for state probabilities of all consecutive deterioration states makes possible computing state durations, which in turns determine shape of the curve. Simple life curves obtained for different maintenance policies are later combined in constructing composite life curves which describe various maintenance scenarios.

For sake of simplicity and consistency, always exactly three deterioration states, or levels, are presented to the end user: minor, medium and major, with adjustable AC ranges. In case of Markov models which have more than three Ds states, the expert decides how to assign Markov states to the three levels when importing the model.

Fig. 6 shows exemplary life curves computed by ARM for typical maintenance situations. In each case the action is delayed for 3 time units (months, for example) and the analysis is performed for a time horizon of 10 time units. In case of failure seen in “Do nothing” action, equipment is repaired and its condition is restored to 85%.

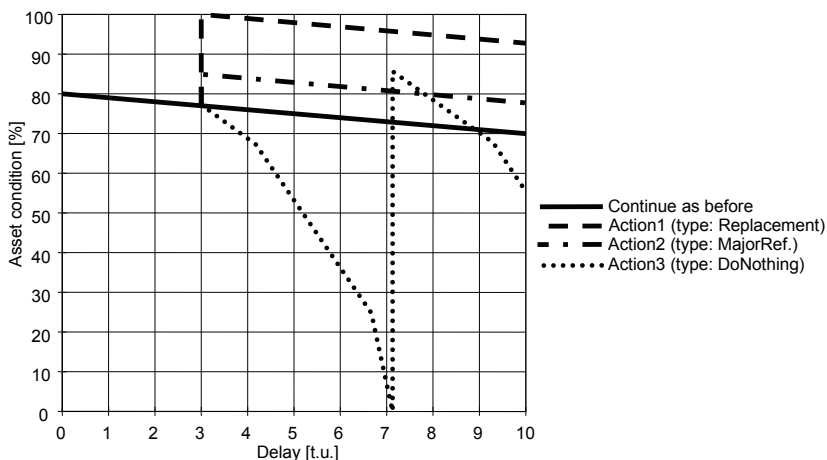


Fig 6. Life curves computed for three different actions (“Action1” ... “Action3”) and compared to the present maintenance policy (“Continue as before”).

4.3 Probability of failure

For a specific action, probability of failure within the time horizon (PoF_{TH}) is a sum of two probabilities: of failure taking place before (PoF_B) and after (PoF_A) the moment of action. It is assumed that failures in these two periods making up the time horizon are independent, so

$$PoF_{TH} = PoF_B + PoF_A - PoF_B \cdot PoF_A.$$

To compute $PoF(T)$ within some time period T , the Markov model for the equipment and the life curve are required. The procedure is as follows:

- (1) For initial asset condition, find from the life curve the current deterioration state DS_n ; compute also state progress (SP, %), i.e. estimate how long the equipment has been in the DS_n state.
- (2) Running FPT analysis on the model, find distributions D_n and D_{n+1} of first passage time from DS_n and DS_{n+1} to the failure state F.
- (3) Taking state progress into account, probability of failure is evaluated as

$$PoF = D_n(T) \cdot (1 - SP) + D_{n+1}(T) \cdot SP$$

For better visualization, rather than finding a single PoF_{TH} value for action defined by the user in input parameters, ARM computes a curve which shows the PoF_{TH} as a function of action delay varying in a range 0 ÷ 200% of user-specified initial value. An example is demonstrated in Fig. 7 for “Do nothing” action (user-defined delay = 3 time units), where also the two probability components PoF_B and PoF_A are shown.

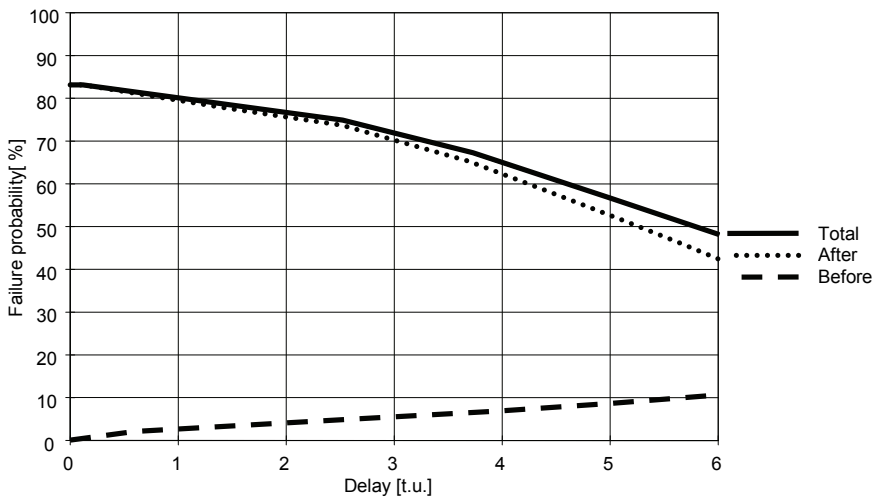


Fig 7. Probability of equipment failure within the time horizon for “Do nothing” action, computed as a function of action delay.

4.4 Cost curves

In many financial evaluations, the costs are expressed as present value (PV) quantities. The present value approach is also used in ARM because maintenance decisions on ageing

equipment include timing, and the time value of money is an important consideration in any decision analysis. The cost difference is often referred to as the Net Present Value (NPV). In the case of maintenance, the NPV can be obtained for several re-investment options which are compared with "Continue as before" policy.

Cost computations involve calculation of the following cost components:

1. cost of maintenance activities,
2. cost of the action selected (e.g. refurbishment or replacement),
3. cost associated with failures (cost of repairs, system cost, penalties, etc.).

To compute the PV, inflation and discount rates are required for a specified time horizon. The cost of maintenance over the time horizon is the sum of the maintenance costs incurred by the original maintenance policy for the duration of the delay period, and the costs incurred by the new policy for the remainder of the time horizon. The costs associated with equipment failure over the time horizon can be computed similarly except that the failure costs before and after the action is multiplied by the respective probabilities of failures (PoF_B and PoF_A), and the two products are added. As in case of probability of failure, ARM presents the end user with a curve which shows the cost as a function of action delay varying in a range 0 ÷ 200% of user-specified value.

5. Conclusions

The purpose of the ARM tool is to help in choosing effective maintenance policy. Based on Markov models representing maintenance actions and deterioration processes, life curves and other reliability parameters can be evaluated. Once a database of equipment models is prepared, the end-user can perform various studies about different maintenance strategies and compare expected outcomes. Since the equipment condition is visualized through the relatively simple concept of a life curve, no detailed expert knowledge about internal reliability parameters or configuration is required.

The system can also automatically adjust the model to requested repair frequencies and thus provides for fully automatic computation of dependability parameters in cases when maintenance policy needs to be modified within some range. This also reduces the model preparation time that requires involvement of the reliability expert and allows for broader range of studies that can be done fully automatically by the end user.

6. References

- Anders G.J. & Endrenyi J. (2004). Using Life Curves in the Management of Equipment Maintenance, *Proceedings of PMAAPS'2004 Conference*, Ames, Iowa, September 2004.
- Anders G.J. & Leite da Silva A.M. (2000). Cost Related Reliability Measures for Power System Equipment, *IEEE Transactions On Power Systems*, Vol. 15, No.2, pp. 654-660.
- Anders G.J. & Maciejewski H. (2006). Estimation of impact of maintenance policies on equipment risk of failure, *Proceedings of International Conference on Dependability of Computer Systems DepCoS - RELCOMEX 2006*, pp. 351-357, ISBN 0-7695-2565-2, Szklarska Poręba, Poland, May 2006, IEEE Computer Society Press.

- Anders G.J. & Sugier J. (2006). Risk Assessment Tool for Maintenance Selection, *Proceedings of International Conference on Dependability of Computer Systems DepCoS – RELCOMEX 2006*, pp. 306-313, ISBN 0-7695-2565-2, Szklarska Poręba, Poland, May 2006, IEEE Computer Society Press.
- Cugnasca P.S., De Andrade M.T.C. & Camargo Jr. J.B. (1999). A fuzzy based approach for the design and evaluation of dependable systems using the Markov model, *Proceedings of 1999 Pacific Rim International Symposium on Dependable Computing*, pp.112-119, ISBN 0-7695-0371-3, Hong Kong, December 1999.
- Duque O. & Morinigo D. (2004). A fuzzy Markov model including optimization techniques to reduce uncertainty, *Proceedings of the 12th IEEE Mediterranean Electrotechnical Conference*, vol.3, pp. 841-844, ISBN 0-7803-8271-4, Dubrovnik, May 2004.
- Endrenyi J. (1978). *Reliability Modeling in Electric Power Systems*, J. Wiley & Sons, Chichester, 1978.
- Endrenyi J., Anders G.J. & Leite da Silva A. M. (1998). Probabilistic Evaluation of the Effect of Maintenance on Reliability - An Application. *IEEE Transactions on Power Systems*, vol. 13, no. 2 (May 1998), pp. 575-583.
- Ge H., Tomasevicz C.L. & Asgarpoor S. (2007). Optimum Maintenance Policy with Inspection by Semi-Markov Decision Processes, *Proceedings of 39th North American Power Symposium*, pp.541-546, ISBN 9-7814-2441-7254, Las Cruces, NM, USA, September/October 2007.
- Hughes D. T. & Russell D. S. (2005). Condition Based Risk Management (CBRM), a Vital Step in Investment Planning for Asset Replacement, *Proceedings of IEE-RTDN Conference*, London UK, February 2005.
- IEEE/PES Task Force on Impact of Maintenance Strategy on Reliability of the Reliability, Risk and Probability Applications Subcommittee (2001). The present status of maintenance strategies and the impact of maintenance on reliability, *IEEE Transactions on Power Systems*, vol. 16, no. 4 (November 2001), pp. 638-646.
- Li Z. & Guo J. (2006). Wisdom about age-aging electricity infrastructure, *IEEE Power and Energy Magazine*, vol.4, no.3 (May-June 2006), pp. 44-51.
- Mohanta D.K., Sadhu P.K. & Chakrabarti R. (2005). Fuzzy Markov model for determination of fuzzy state probabilities of generating units including the effect of maintenance scheduling, *IEEE Transactions on Power Systems*, vol.20, no.4 (November 2005), pp. 2117-2124.
- Singh C. & Billinton R. (1977). *System Reliability Modeling and Evaluation*, London, Hutchinson Educational Publishers, 1977 [Online]. Available: <http://www.ece.tamu.edu/People/bios/singh/sysreliability>.
- Sugier J. & Anders G.J. (2007). Modeling Changes in Maintenance Activities through Fine-Tuning Markov Models of Ageing Equipment, *Proceedings of International Conference on Dependability of Computer Systems DepCoS – RELCOMEX 2007*, pp. 336-343, ISBN 0-7695-2850-3, Szklarska Poręba, Poland, June 2007, IEEE Computer Society Press.
- Tomasevicz C.L. & Asgarpoor S. (to be published). Optimum Maintenance Policy Using Semi-Markov Decision Processes, *Electric Power Systems Research*, to be published.

Computational Experience in Methods for Finding Tight Lower Bounds for the Sparse Travelling Salesman Problem

Fredrick Mtenzi

Dublin Institute of Technology

School of Computing, DIT Kevin Street, Dublin 8

Ireland

1. Introduction

The Sparse Travelling Salesman Problem (Sparse TSP) which is a variant of the classical Travelling Salesman Problem (TSP) is the problem of finding the shortest route of the salesman when visiting cities in a region making sure that each city is visited at least once and returning home at the end. In the Sparse TSP, the distance between cities may not obey the triangle inequality; this makes the use of algorithms and formulations designed for the TSP to require modifications in order to produce near-optimal results.

When solving the Sparse TSP, our main interest is in computing feasible tours at a reasonable computational cost. In addition because it is not possible to get an optimal solution most of the time, we would like to have some guarantee on the quality of the tours (solutions) found. Such guarantees can most of the time be provided if a lower bound on the length of a shortest possible tour is known.

It is also the case that most algorithms for finding exact solutions for large Standard TSP instances are based on methods for finding upper and lower bounds and an enumeration scheme. For a given instance, lower and upper bounds are computed. In most cases, these bounds will not be equal, and therefore, only a quality guarantee for the feasible solution can be given, but optimality cannot be proved. If the upper and lower bounds coincide, a proof of optimality is achieved.

Therefore, the determination of good tours and derivation of tight lower bounds can be keys to a successful search for optimal solutions. Whereas there have been a lot of work and progress in designing heuristic methods to produce upper bound, the situation for lower bounds is not as satisfying.

In general for the Standard TSP, lower bounds are obtained by solving relaxations of the original problem in the sense that one optimizes over some set containing all feasible

solutions of the original problem as a (proper) subset. This then means, for example, that the optimal solution of the relaxed problem gives a lower bound for the value of the optimal solution of the original problem. In practice, the methods usually used for computing lower bound for the Standard TSP are the Held-Karp lower bound (Johnson et al., 1996) and Lagrangian relaxation (Reinelt, 1994).

Since the Sparse TSP is an NP-Hard combinatorial optimization problem as per Fleischmann (Fleischmann, 1985), the standard technique to solve it to optimality is based on an enumeration scheme which for large problems is computationally expensive. Therefore a natural way is to use the Sparse TSP heuristics to obtain a near-optimal solution. Solutions obtained by heuristics for the Sparse TSP provide the upper bounds. Heuristics produce feasible solutions but without any quality guarantees as to how far off they may be from the optimal feasible solution. In order to be able to assess the performance of heuristics we need to find the lower bound of the problem.

Therefore, in this chapter we are interested in exploring methods for computing tight lower bounds for the Sparse TSP. This is the case because we do not have the luxury of comparing with what other researchers have done, since most of the work in the TSP has been focused on the Standard TSP. For example, in the Standard TSP there are sample instances with optimal solutions provided in the TSPLIB for most of the problems (see (Reinelt, 1992)). The results given in TSPLIB include a provable optimal solution if available or an interval given by the best known lower bound and upper bound. As far as we are aware there are no such benchmark results for the Sparse TSP which is studied in this chapter.

A lower bound gives us the quality guarantee of the near-optimal solution obtained by using heuristic methods. The most widely used procedure for finding the lower bound for the Standard TSP is the Held and Karp lower bound (Held & Karp, 1970). Johnson et al (Johnson et al., 1996) provide empirical evidence in support of using the Held and Karp (HK) lower bound as a stand-in for the optimal tour length when evaluating the quality of near-optimal tours. They show that for a wide variety of randomly generated instances the optimal tour length averages less than 0.8% over the HK lower bound, and for the real world instances in TSPLIB the gap is always less than 2%. A tight lower bound for the Sparse TSP will play a key role in developing and assessing the performance of the Sparse TSP heuristic methods.

Definitions

A *relaxation* of an optimization problem P is another optimization problem R , whose set of feasible solutions \mathfrak{R} properly contains all feasible solutions P of P . The objective function of R is an arbitrary extension on \mathfrak{R} of the objective function of P . Consequently, the objective function value of an optimal solution to R (minimisation case) is less than or equal to the objective function value of an optimal solution to P . If P is a hard combinatorial problem and R can be solved efficiently, the optimal value of R can be used as a lower bound in an enumeration scheme to solve P . The closer the optimal value of R to the optimal value of P , the more efficient is the enumeration algorithm.

A *lower bound* of the TSP is the value obtained by solving a relaxation of the original problem or by using heuristics. Its value is in most cases less than the optimal value of the original problem, it is equal to optimal value when the value of lower bound is equal to the value of the upper bound.

In this chapter, we propose and give computational experience on two methods of finding lower bounds for the Sparse TSP, the chapter is organised as follows. In section 2, we give the background and related research in the Travelling Salesman Problem and specifically how this work relates to the Sparse TSP. In section 3, we discuss methods for finding the lower bound of the Sparse TSP, and give the formulation and relaxation for the Linear Programming relaxation of the Sparse TSP. Further, we introduce the Arc-cutset Partial Enumeration Strategy as a strategy for finding the lower bound of the Sparse TSP. Finally, section 4 gives the conclusions and summary.

2. Related works

The Standard TSP has been an area of intensive research since the late 1950's as demonstrated in (Lawler et al., 1985). The first major breakthrough came in 1954, in a seminal paper by (Dantzig et al., 1954) where a description of a method for solving the Standard TSP was given and its power illustrated by solving an instance of 49 cities. From there on there has been a lot of research published on the Standard TSP and its variants.

Most of the methods used for solving the Standard TSP to optimality are of the branch and bound variety where, at each node of the branch and bound tree, lower bounds are computed by solving related problems which are relaxations of the original TSP (see (Camerini et al., 1975), (Gabovich, 1970), Held and Karp (Held & Karp, 1970), Padberg and Hong (Padberg & Hong, 1977), and Rubinshtein (Rubinshtein, 1971)). As for all branch and bound methods, the quality of the computed lower bounds at each node has much greater influence on the effectiveness of the algorithm than any branching rules that may be used to generate the subproblems during the search. Branch and Bound techniques have been used successfully in optimization problems since the late 1950's. Several different branch and bound algorithms are possible for the travelling salesman problem. A survey of these is given in Bellmore and Nemhauser (Bellmore & Nemhauser, 1968). A variation of Branch and Bound techniques using cutting plane techniques called Branch-and-Cut by Padberg and Rinaldi((Padberg & Rinaldi, 1989), (Padberg & Rinaldi, 1991)) is a much more powerful technique for computing solutions for the Standard TSP.

Held and Karp (Held & Karp, 1970), (Held & Karp, 1971) pioneered an iterative approach which uses a technique called Lagrangian Relaxation to produce a sequence of connected graphs which increasingly resemble tours. This technique is based on the notion of a 1-tree and the bound generated is called the Held-Karp lower bound (HK lower bound). Formulating the Standard TSP as an integer linear programming problem (see Dantzig et al (Dantzig et al., 1954), Miller et al (Miller et al., 1960), Fox et al (Fox et al., 1980), and Claus (Claus, 1984) and systematically solving its relaxations is another way of obtaining a lower bound for TSP. Bounds from the solutions of the assignment problem, the matching problem, and the shortest n-path problem have also been suggested and explored by

Christofides (Christofides, 1979) who also gave a brief survey and references. Johnson et al (Johnson et al., 1996) use the HK lower bound as a stand-in for the optimal tour length when evaluating the quality of near-optimal tours in a number of their studies in which they solve problems of up to one million cities.

When the theory of NP-completeness was developed, the Standard TSP was one of the problems shown to be NP-hard by Karp in (Karp, 1972). This remains true even when additional assumptions such as the triangle inequality or Euclidean distances are involved (see Garey (Garey, 1976)). These results imply that a polynomially bounded exact algorithm for the TSP is unlikely to exist. Nevertheless ingenious algorithms for the TSP have been proposed by Little et al (Little et al., 1963), Held and Karp (Held & Karp, 1970), (Held & Karp, 1971), Miliotis (Miliotis, 1976), Crowder and Padberg (Crowder & Padberg, 1980). Over the past four decades the TSP has remained the prototype of a “hard” combinatorial problem. Since the introduction of NP-completeness theory in 1971 and the subsequent inclusion of the TSP in the NP-complete class, some of the mystery has gone out of the TSP. A complete treatment of NP-completeness theory and its relationship to the TSP is given in Garey and Johnson (Garey & Johnson, 1979).

The NP-complete results have given a new impetus to heuristic methods for solving “hard” combinatorial problems. Due to the difficulty of the TSP, many heuristic procedures have been proposed and developed. These heuristics have been compared analytically in the ground-breaking paper of Rosenkrantz, Stearns and Lewis (Rosenkrantz et al., 1977) by studying their worst-case behaviour. Alternatively, computational experimentation may be used to compare the performance of these heuristics, as in the work of Golden et al (Golden et al., 1980) and Stewart (Stewart Jr, 1987). For example, Stewart (Stewart Jr, 1987) describes a number of new algorithms designed specifically to perform well on Euclidean problems.

Much work has been done on fast heuristic algorithms for the Standard TSP. There is a trade-off between the speed of an algorithm and its ability to yield tours which are close to the optimal. The following studies show an enormous interest which researchers have shown to heuristic algorithms. These studies include those of Adrabinski and Syslo (Adrabinski & Syslo, 1983), Golden et al (Golden et al., 1980), Johnson and McGeoch (Johnson & McGeoch, 1995), Bentley (Bentley, 1993), Reinelt (Reinelt, 1992), and Jünger et al (Jünger et al., 1995). There are two broad classes of heuristics for the travelling salesman problem constructive and improvement heuristics. A typical tour construction heuristic method starts with a node and adds others one by one until the tour is complete. Many other variants are described in Bellmore and Nemhauser (Bellmore & Nemhauser, 1968), Rosenkrantz et al (Rosenkrantz et al., 1977), Johnson (Johnson, 1990), Golden et al (Golden et al., 1980), Golden and Stewart (Golden & Stewart, 1985), Gendreau et al. (Gendreau et al., 1992), and Bentley (Bentley, 1993).

The arc-exchange strategy was first applied to the Standard TSP by Croes (Croes, 1958). He suggested the 2-optimal algorithm for the symmetric TSP. About the same time and independently, a 3-optimal strategy was suggested by Bock in (Bock, 1958). However, it was Lin in (Lin, 1965) who truly established through extensive empirical study that the 3-optimal algorithm was indeed an excellent approximation algorithm for the Standard TSP. A

substantial improvement in implementation of the 3-optimal (in general, r -optimal where $r \geq 2$) algorithm was given by Christofides and Eilon in (Christofides & Eilon, 1979). Lin and Kernighan (Lin & Kernighan, 1973) added another level of sophistication to the r -optimal algorithm. Instead of having a fixed value of 2 or 3, r was allowed to vary. Their paper is a classic paper on the local search heuristics for the Standard TSP. In particular, Lin-Kernighan(LK) and its variants are widely recognized as the best (most accurate) local search heuristic for the TSP.

The ejection chains method introduced by Glover (Glover, 1992) have been used to generate compound neighbourhood structures for the Standard TSP by Rego (Rego, 1998). It should be noted that this neighbourhood has been used to solve problems taken from Travelling Salesman Problem LIBrary(TSPLIB) and performed better than the best iterated LK, but it took more time.

In order for the heuristics algorithms to work well efforts should be spent on designing efficient data structures. Data structures play an important role in the design and implementation of the Standard TSP algorithms. The following data structures have been used in the Standard TSP study, the k - d tree of Bentley discussed and used in Bentley (Bentley, 1993), (Bentley, 1975), (Bentley, 1990), and (Bentley, 1990). Fredman et al (Fredman et al., 1995) discuss and give implementation details on the following data structure for the Standard TSP, *array-based* and *splay trees*.

We have witnessed remarkable progress in Mathematical Programming, Polyhedral Theory, and significant advances in computer technology that have greatly enhanced the chances of finding exact solutions to reasonably large combinatorial optimization problems. Nevertheless, these problems still pose a real challenge in the sense that it is hard to solve realistically large problem instances in reasonable computational times. This challenge has led to the development of many approximate algorithms and has made the science of heuristics a fast growing area of research. For more details on combinatorial optimization problems we refer the reader to Nemhauser and Wolsey (Nemhauser & Wolsey, 1988), Papadimitriou and Steiglitz (Papadimitriou & Steiglitz, 1982), Garey and Johnson (Garey & Johnson, 1979).

Metaheuristics are a class of approximate solution methods, which have developed dramatically since their inception in the early 1980s. They are designed to solve complex optimization problems where classical heuristics and other optimization methods have failed to be effective and efficient. A *metaheuristic* can be defined as an iterative generation process which guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space, learning strategies are used to structure information in order to find efficiently near-optimal solutions (see Osman (Osman, 1995), and Osman and Kelly (Osman & Kelly, 1996)). Hence, metaheuristics include, but are not limited to: Constraint Logic Programming; Genetic Algorithms; Greedy Random Adaptive Search Procedures; Neural Networks; Non-monotonic search strategies; Problem and heuristic search-space; Simulated Annealing; Tabu Search; Threshold Algorithms and their hybrids. These techniques are based on concepts borrowed from biological evolution, intelligent problem solving, Mathematical and Physical Sciences, Nervous System, and

Statistical Mechanics. For details we refer the reader to Reeves (C.R.Reeves, 1993; Reeves, 1993). However, Meta-heuristics have not been very successful in solving the Standard TSP according to Johnson and McGeoch (Johnson & McGeoch, 1995). They have been very effective in other areas such as industrial applications and communications networks.

The solution produced by using local search procedures can be very far off from optimality. In order to avoid such disadvantages while maintaining the simplicity and generality of the approach, the following concepts which form the basis of most metaheuristics are considered see Reeves (Reeves, 1993). Start from good initial solutions which can be generated intelligently using a greedy random adaptive search designed by Feo and Resende (Feo & Resende, 1995), or space-search methods in Storer, Wu and Vaccari (Storer et al., 1995). Use the learning strategies of neural networks discussed in Sharda (Sharda, 1994), and Tabu search in Glover (Glover, 1995) that gather information during the algorithms execution in order to guide the search to find possibly better solutions. Employ non-monotonic search strategies that sample/accept neighbours based on hybrid modifications of simulated annealing and Tabu Search methods or others see (Glover (Glover, 1995), (Glover, 1995)), (Hu, Khang and Tsao (Hu et al., 1995)), (Osman (Osman, 1993), (Osman, 1995), (Osman, 1995)) and (Osman and Christofides (Osman & Christofides, 1994)).

Sahni and Gonzales in (Sahni & Gonzales, 1976) showed that if a triangle inequality is not satisfied, the problem of finding an approximate solution to the TSP within any fixed bound ratio of the optimum is as hard as finding an exact solution. A comprehensive treatment of various aspects of the TSP can be found in the collection of research papers in Lawler et al (Lawler et al., 1985).

3. Methods for finding Lower bound for the Sparse TSP

The standard technique for obtaining lower bounds on the Standard TSP is to use a relaxation that is easier to solve than the original problem. These relaxations can have either discrete or continuous feasible sets. Several relaxations have been considered over years for the Standard TSP. We are going to introduce modifications to these relaxations, so that they can be used to find lower bounds for the Sparse TSP at a reasonable computational effort.

We can illustrate the relaxation of an optimization problem by figure 1 in which the region ABCD is the feasible region of the relaxed problem, while the region MNOPQRS is the feasible region of the original "true" problem, which in this case happens to contain integer points. Solution ABCD may be obtained when we relax integrality constraints. Relaxation of the combinatorial optimization has become so popular because in most cases the problems can be solved with reasonable computational effort, if not easier than the original problem.

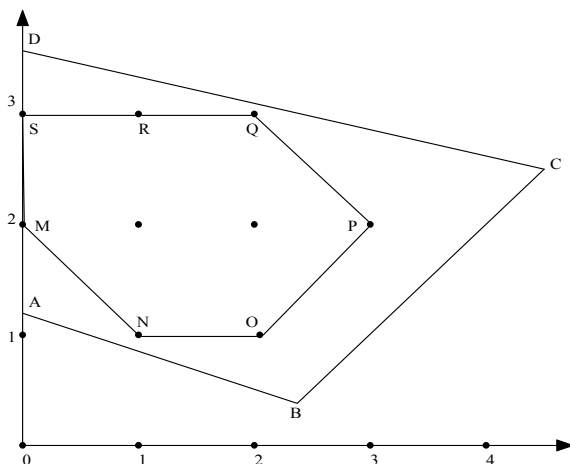


Fig. 1. Region MNOPQRS shows a feasible solution to the original “true” problem while region ABCD shows a feasible solution to the relaxed problem

The HK lower bound is the solution to the LP relaxation of the integer programming formulation of the Standard TSP (see Dantzig et al (Dantzig et al., 1954), Reinelt (Reinelt, 1994) and Johnson et al (Johnson et al., 1996)). That is, it is the Integer Linear Programming with the integrality constraints relaxed. The HK lower bound provides a very good estimate of optimal tour length for the Standard TSP. This measure has enormous practical value when evaluating the quality of near optimal solutions for large problems where the true optimal solutions are not known or are computationally expensive to find. The HK lower bound has been used as a stand-in for the optimal tour length when evaluating the quality of near-optimal tours in a lot of studies (for example, in Johnson et al (Johnson et al., 1996)).

Although the HK lower bound can be evaluated exactly by Linear Programming techniques, code for doing this efficiently for problems larger than a few hundred cities is not readily available or easy to produce (see Valenzuela and Jones (Valenzuela & Jones, 1996)). In addition linear programming implementations (even efficient ones) do not scale well and rapidly become impractical for problems with many thousands of cities. To be able to find the HK lower bound, a procedure for finding violated inequalities must be provided. This is not a simple matter of automatically generating violated inequalities. It is because of the above mentioned difficulties that most researchers have preferred to use the iterative estimation approach for finding lower bound for the Standard TSP proposed by Held and Karp (Held & Karp, 1970), (Held & Karp, 1971). In this chapter we use this method and modify it to solve the Sparse TSP problems.

3.1 The LP Relaxations for the Sparse TSP

The formulation for the Integer Linear Programming (ILP) Sparse TSP is given as:

$$\min_x \sum_{i=1}^N \sum_{j=1}^N c_{ij} x_{ij} \tag{1.1}$$

Subject to

$$\sum_{j:(i<j)\in A} x_{ij} + \sum_{j:(j<i)\in A} x_{ji} = 2m_i, \text{ for all } i \in N \quad (1.2)$$

$$\sum_{i \in S, j \in N-S, i < j} x_{ij} + \sum_{i \in S, j \in N-S, j < i} x_{ji} \geq 2, \text{ for all } S \subset N, S \neq \emptyset \quad (1.3)$$

$$m_i \geq 0 \text{ for all } i \in N \quad (1.4)$$

$$x_{ij} \in \{0,1\} \text{ for all } i, j = 1, \dots, N \quad (1.5)$$

By relaxing the integrality constraint (1.5) we get the Linear Programming (LP) relaxation for the ILP Sparse TSP where equations (1.1, 1.2, 1.3 and 1.4) remains the same and constraint (1.5) becomes,

$$x_{ij} \geq 0 \text{ for all } i, j = 1, \dots, N \quad (1.6)$$

Note: that any integral solution to the LP relaxation is a tour.

We solved the LP relaxation for the modified ILP Sparse TSP formulation problems, results given in figure 7 were obtained by using violated arc-cutset constraints.

3.2 The LP relaxation for the EFF for the Sparse TSP

From the single commodity flow formulation, we present its modification which we call the Embedded Flow Formulation (EFF) for the Sparse TSP. This formulation involves a polynomial number of constraints, even though the number of variables is increased considerably. The EFF for the Sparse TSP is given as:

$$\min_x \sum_{i=1}^N \sum_{j=1}^N c_{ij} x_{ij} \quad (1.7)$$

Subject to

$$\sum_{j:(i<j)\in A} x_{ij} + \sum_{j:(j<i)\in A} x_{ji} = 2m_i, \text{ for all } i \in N \quad (1.8)$$

$$y_{ij} - (n-1)x_{ij} \leq 0, \text{ for all } (i, j) \in A \quad (1.9)$$

$$\sum_{j:(i<j)\in A} y_{ij} - \sum_{j:(j<i)\in A} y_{ji} = -1, \text{ for all } i = 2, 3, \dots, n \quad (1.10)$$

$$\sum_{i \in A} (y_{ii} - y_{i1}) = n-1 \quad (1.11)$$

$$m_i \geq 0, \text{ for all } i \in N \quad (1.12)$$

$$x_{ij} = 0 \text{ or } 1, \text{ for all } (i, j) \in A \quad (1.13)$$

$$y_{ij} \geq 0, \text{ for all } (i, j) \in A \quad (1.14)$$

The LP relaxation for the EFF for the Sparse TSP formulation is obtained by relaxing the integrality constraint (1.13). All other equations remain the same except constraint (1.13) changes to:

$$x_{ij} \geq 0, \text{ for all } (i, j) \in A \quad (1.15)$$

It was interesting to know how close the lower bound Z_{LB} obtained by solving the subtour relaxation is to the length of an optimal tour Z_{opt} . Worst-case analysis of HK lower bound by

Wolsey (Wolsey, 1980) and Shmoy and Williamson (Shmoys & Williamson, 1990) show that for any cost function C satisfying the triangle inequality, the ratio Z_{LB}/Z_{opt} is at least $2/3$. The $2/3$ lower bound is not shown to be tight and actually it is conjectured by Goemans (Goemans, 1995) that $(Z_{LB}/Z_{opt}) \geq 3/4$. Our computational results show that for many instances the above ratio is very close to 1.

The results we obtained for the EFF for the Sparse TSP are presented in figure 8. In general the LP relaxation is not equal to the minimum tour length but it is very close. LP relaxation for the ILP Sparse TSP gives a much tighter lower bound than the LP relaxation for the EFF for the Sparse TSP and requires less iterations.

3.3 An Arc-cutset Partial Enumeration Strategy (APES)

In this section we are proposing a strategy for quickly generating some of the violated arc-cutset constraints which we call an *Arc-cutset Partial Enumeration Strategy* (APES). The APES is based on the following observation, using the formulation for the ILP Sparse TSP, we can drop the connectivity constraints. When the resulting formulation is solved the solution produces a lot of disconnected components, all have at least two nodes connected by two arcs. That is to say each component is a subtour, and the obtained solution is not a tour.

These components needed to be connected with other components to produce a single connected component, a tour. To be able to achieve this, we generated an arc-cutset constraint for each component. In other words we generated an arc-cutset constraint for each arc in the graph. This approach is reasonable to the Sparse TSP because the number of arcs m in the sparse graph is $O(n)$ as opposed to $O(n^2)$ in the complete graph. Arc cutset identified between two components resulted in those components being connected plus several others. Therefore, even if when the APES is applied at first, there are may be say five disconnected components, it does not mean that five arc cutset will have to be identified. Our computational study showed that the number of arc cutset to be identified is always less than five. However, we were not able to come up with the relationship between the size of the problem and the number of arc cutset to be identified before obtaining a tour.

The arc-cutset constraints generated this way are all valid inequalities. Naddef and Rinaldi (Naddef, 1992), Cornuéjols et al (Cornuéjols et al., 1985), and Swamy and Thulasiraman (Swamy & Thulasiraman, 1981) have shown the validity of the arc-cutset constraints. They say that once the components are connected then the violated arc-cutset constraints are valid inequalities.

Our algorithm for the APES is given below.

```

An Arc-cutset Partial Enumeration Strategy algorithm
Step 1: Formulate the problem using evenness condition
        Constraints and integrality constraints only.
        Let nodes be the number nodes in the starting path
        Let nodesv be the number of nodes to be visited
        Let k := 2
Step 2: For nodes := k to n do
        For nodesv := 3 to n - k do
            List all arcs incident to the path
            with end node 1 and nodesv
            add the arc-cutset constraint to the formulation
        EndFor
    EndFor
Step 3: Solve the new formulation using any LP solver
Step 4: stop

```

In forming the arc-cutset constraints, we first used a subtour component consisting only of two end nodes i and j with (i,j) as a connected component. The violated arc-cutset constraints were constructed by listing all arcs incident to node i and node j to form one violated arc-cutset constraint, i.e., all arcs incident with a subtour component. The arc connecting node i and node j was not included in the arc-cutset constraints. Figure 2 shows how using component (i,j) arc-cutset constraints was formed. This is what takes place in step 2 of the APES algorithm.

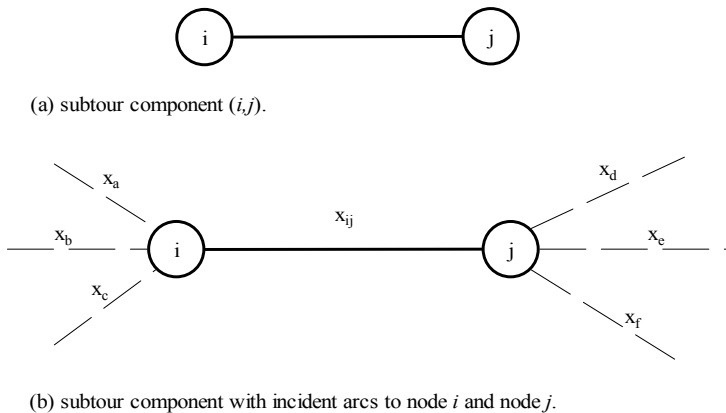


Fig. 2. Formulation of the arc-cutset constraint

$$x_a + x_b + x_c + x_d + x_e + x_f \geq 2 \quad (1.16)$$

The arc-cutset constraints which are generated by the APES are used to connect components. Since the APES starts by using the evenness condition constraints and integrality constraints, while omitting the connectivity constraints. For example the twenty nodes problem shown in figure 3, is used to demonstrate how the APES works.

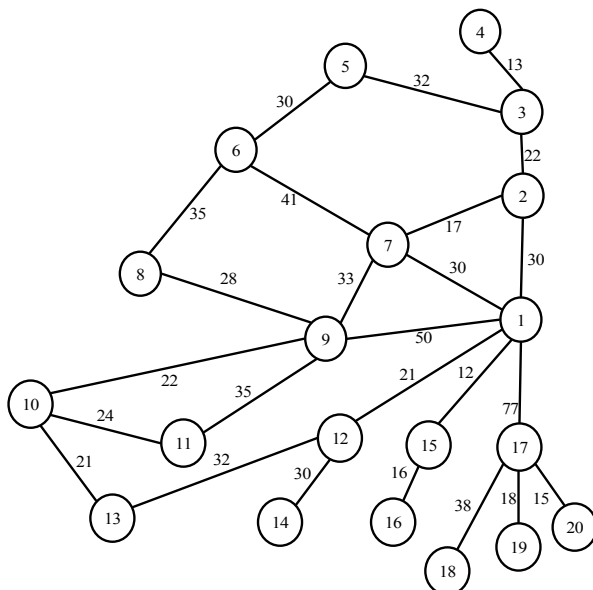


Fig. 3. A twenty nodes problem to demonstrate how APES works

Solving the twenty nodes problems before adding the violated arc-cutset constraints generated by the APES gives the disconnected tours illustrated in figure 4 and whose objective function is 524.

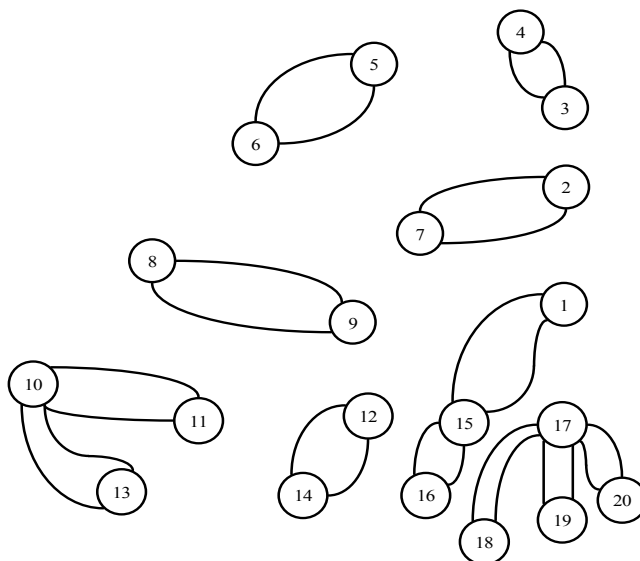


Fig. 4. A twenty nodes problems with disconnected subtour components

After adding the violated arc-cutset constraints generated by the APES we got a tour as illustrated in figure 5 and its objective function was 765, which in this case happened to be the optimal tour.

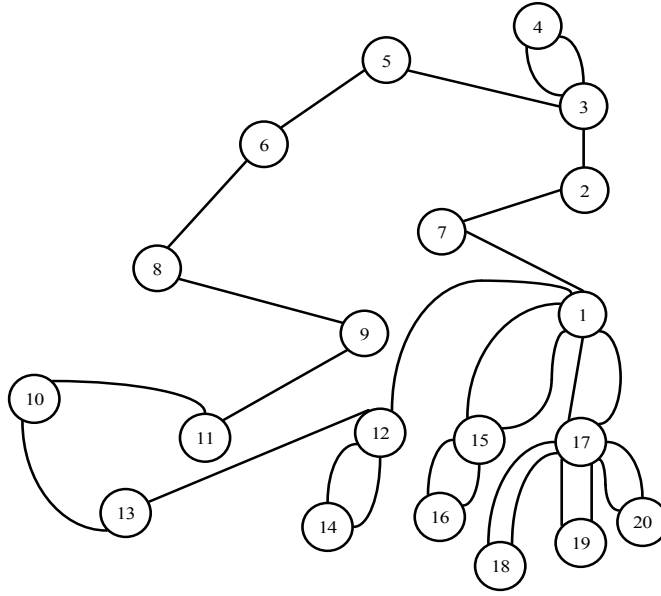


Fig. 5. A twenty nodes problem tour

For all the problems reported in this study, we used the APES to generate violated arc-cutset constraints from each arc. This strategy was used to ensure that disconnected components were connected to form a tour. The APES was able to produce optimal tours for all small problems we solved of up to 1000 nodes. It is interesting to note how the algorithm performed in some graphs. For example, in the case of the 30 nodes problem we had to add eleven violated arc-cutset constraints before getting a tour, while we had to add only two violated arc-cutset constraints for the 67 nodes problem to get a tour. As pointed out earlier the number of violated arc-cutset required to be identified seem to be a function of the nature of the graph and not its size.

We then extended this technique of identifying violated arc-cutset constraints. These new violated arc-cutset constraints were formed by visiting a path of three or more nodes together. The first violated arc-cutset constraint was formed by visiting any three nodes which formed a path. When forming these constraints, we included all arcs which were incident to nodes 1 or 2 or 3 and ignore arcs connecting nodes 1, 2, and 3. The next constraint was formed by adding the fourth node to the path and the violated arc-cutset constraint was identified by listing all nodes incident to the path consisting of four nodes ignoring the arcs which form the path. The process continued until we got a tour. Figure 6 shows how these violated arc-cutset constraints were formed.

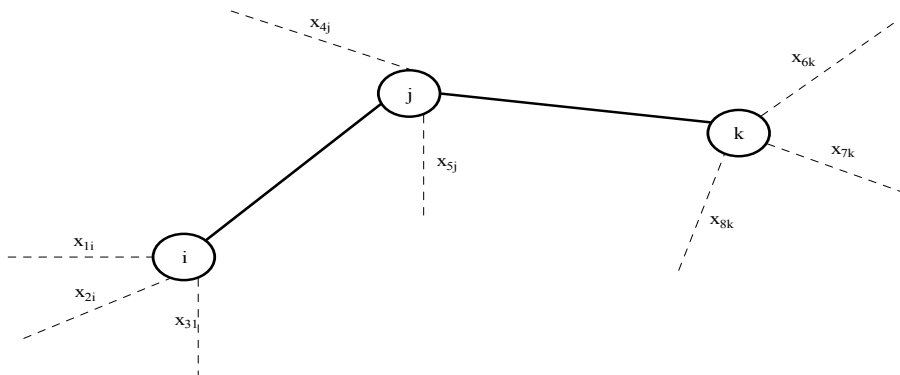


Fig. 6. An example of how the extended technique for identifying violated arc-cutset constraints works

From figure 6 the following violated arc-cutset constraint (1.17) will be formed:

$$x_{1i} + x_{2i} + x_{3i} + x_{4j} + x_{5j} + x_{6k} + x_{7k} + x_{8k} \geq 2 \tag{1.17}$$

The results in figure 6 shows how the APES method performed when the starting path visited 3, 4, ... , 10 nodes together. In other words at first the starting path had 3 nodes and we extended the path by adding one node at a time. The second time the starting path had 4 nodes and we extended the path by adding one node at a time. We continued increasing the number of nodes in a starting path, until at last our starting path had 10 nodes to start with. We got in a good number of cases substantial improvements in the lower bound as we increased the number of nodes in the starting path. However, as the number of nodes in the starting path went beyond 10 we got marginal improvement.

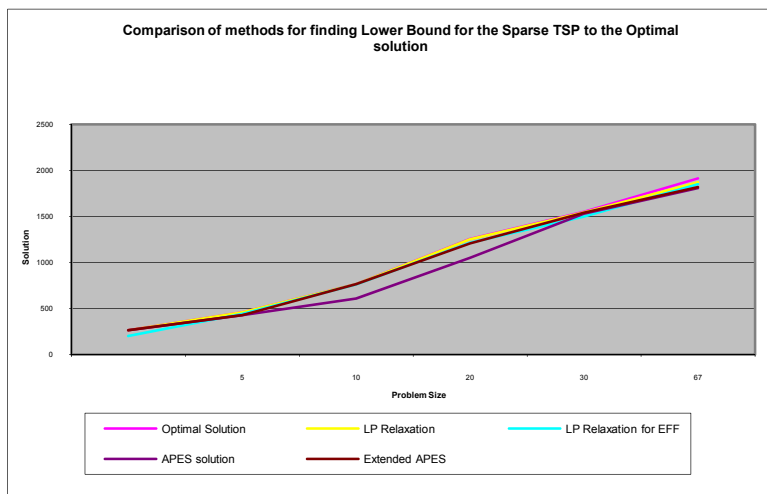


Fig. 7. Comparison of methods for finding Lower Bound for the Sparse TSP to the optimal solution

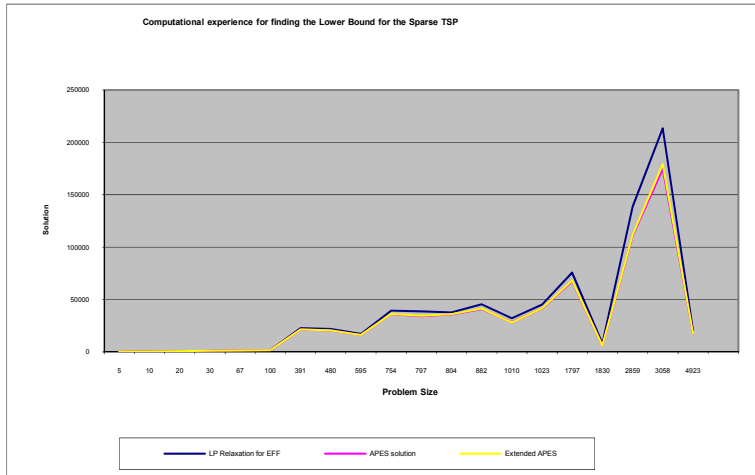


Fig. 8. Computational experience for finding the Lower Bound for the Sparse TSP

The order of complexity of our APES is $O(n)$. This is far less than the connectivity constraints with the order of complexity of $O(n^2)$. This makes our strategy much easier to use.

4. Summary and future work

When optimization problems arise in practice we want to have confidence in the quality of the solutions. Quality guarantees are required because in most cases and especially for large problems, it is not always possible to find an optimal solution. Quality guarantees become possible by being able to compute tight lower bounds at a reasonable computational cost. In this chapter we have proposed two methods for finding tight lower bound for the Sparse TSP using the modified Linear Programming relaxation method and the Arc-cutset Partial Enumeration Strategy.

When the LP relaxation method is used to find a lower bound for the ILP Sparse TSP, finding arc-cutset constraints is a headache especially for large problems. There are procedures for identifying violated arc-cutset constraints automatically in practice, such as the separation routines. These procedures are computational expensive and therefore were not used in this study.

The Arc-cutset Partial Enumeration Strategy proposed is a simple and fast way of getting a lower bound without spending time in a separation algorithm. Computational results show that the lower bounds obtained by using this method are as comparable as those obtained using Separation algorithms.

A lower bound on the optimal value (assuming a minimization problem) is obtained from a relaxation of the integer program. In the past fifteen years attention has shifted from

Lagrangian Relaxation to Linear Programming relaxation, since the latter type of relaxation can be strengthened more easily by using cutting planes. Combining cutting planes and Lagrangian relaxation usually causes convergence problems as discussed by Aardal et al (Aardal et al., 1997). Utilising various modified LP relaxation to suit the Sparse TSP has given us the tightest lower bound of all lower bound techniques we have discussed in this chapter.

5. References

- Aardal, K. et al. (1997). A decade of Combinatorial Optimization, Working paper, Uetrecht University, Computer Science Department, Netherlands.
- Adrabinski, s. A. & Syslo, M. M. (1983). Computational experiments with some approximation algorithms for the travelling salesman problem. *Zastos. mat.*, 18, 1, 91-95.
- Bellmore, M. & Nemhauser, G. L. (1968). The Traveling salesman problem: A survey. *Operations Research*, 16, 538-558.
- Bentley, J. J. (1993). Fast Algorithms for Geometric Traveling Salesman Problems. *ORSA Journal on Computing*, 4, 4, 387-411.
- Bentley, J. L. (1990). Experiments on traveling salesman heuristics, *The 1st Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, PA.*, 91-99.
- Bentley, J. L. (1990). K-d trees for semidynamic point sets, *The 6th Annual Symposium on Computational Geometry, ACM, New York*, 187-197.
- Bentley, J. L. (1975). Multidimensional binary search trees used for associative search. *Comm. ACM*, 18, 309-517.
- Bock, F. (1958). An algorithm for solving Traveling Salesman and related network optimization problems, *inproceedings 14th National meeting of the ORSA*, St. Louis, Mo., St. Louis, Mo.
- Reeves C.R. (1993). *Modern Heuristic Techniques for Combinatorial Problems*, McGraw-Hill Book Company.
- Camerini, P. et al. (1975). Travelling salesman problem: Heuristically guided search and modified gradient techniques.
- Christofides, N. (1979). The Travelling Salesman Problem, *Combinatorial Optimization*, N. Christofides, Mingozzi, A., Toth, P. and Sandi, C., 131-149, John Wiley and Sons. John Wiley and Sons.
- Christofides, N. & Eilon, S. (1979). Algorithm for large-scale Traveling Salesman Problems. *Operational Research Quarterly*, 23, 4, 511-518.
- Claus, A. (1984). A new formulation for the traveling salesman problem. *SIAM Journal of Algebraic Discrete Methods*, 5, 21-25.
- Cornuéjols, G. et al. (1985). The travelling salesman problem on a graph and some related integer polyhedra. *Mathematical Programming*, 33, 1-27.
- Croes, G. A. (1958). A method for solving traveling salesman problems. *Operations Research*, 6, 791-812.
- Crowder, H. & Padberg, M. (1980). Solving large-scale symmetric travelling salesman problem to optimality. *Management Science*, 26, 495-509.
- Dantzig, G. et al. (1954). Solution of a large-scale Traveling-Salesman Problem. *Operations Research*, 2, 393-410.

- Feo, T. A. & Resende, M. G. C. (1995). Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6, 109-118.
- Fleischmann, B. (1985). A cutting plane procedure for travelling salesman problem on road networks. *European Journal of Operational Research*, 21, 307-317.
- Fox, K. et al. (1980). An n-constraint formulation of the (time-dependent) traveling salesman problem. *Operations Research*, 28, 1018-1021.
- Fredman, M. L. et al. (1995). Data Structures for Travelling Salesmen. *Journal of Algorithms*, 18, 432-475.
- Gabovich, E. J. (1970). The small travelling salesman problem. 19, 27-51.
- Garey, M. (1976). Some NP-complete geometric problems, *inproceedings Proceedings of the 8th SIGACT Symp. on the theory of computing*, 10-22.
- Garey, M. R. & Johnson, D. S. (1979). *Computers and Intractability: A Guide to the theory of NP-Completeness*, W.H. Freeman, San Francisco.
- Gendreau, M. et al. (1992). New Insertion and Postoptimization procedure for the traveling salesman problem. *Operations Research*, 40, 1086-1094.
- Glover, F. (1995). Tabu thresholding: improved search by non-monotonic trajectories. *ORSA Journal on computing*, 7, 426-436.
- Glover, F. (1995). Tabu search fundamentals and uses, Graduate School of Business, University of Colorado, Boulder.
- Glover, F. (1992). New ejection chain and alternating path methods for the traveling salesman problems, *Operations research and computer science: new developemnts in their interfaces*, O. Balci, Sharda, R. and Zenios, S. A., 27-49, Pergamon Press, Oxford., Pergamon Press, Oxford.
- Goemans, M. X. (1995). Worst-case Comparison of valid Inequalities for the TSP. *Mathematical Programming*, 69, 335-349.
- Golden, B. et al. (1980). Approximate travelling salesman algorithms. *Operations Research*, 28, 3, 694-711.
- Golden, B. L. & Stewart, W. R. (1985). Empirical analysis of heuristics, *The Traveling Salesman Problem*, E. L. Lawler, Lenstra, J. K., Kan, A. H. G. R. and Shmoys, D. B., 207-249, John Wiley and Sons Ltd., John Wiley and Sons Ltd.
- Held, M. & Karp, R. M. (1970). The Travelling Salesman Problem and Minimum spanning Trees, Part I. *Operations Research*, 18, 1138-1162.
- Held, M. & Karp, R. M. (1971). The Travelling Salesman Problem and Minimum spanning Trees, Part II. *Mathematical Programming*, 1, 6-25.
- Hu, T. C. et al. (1995). Old bachelor acceptance: A new class of non-monotonic threshold accepting methods. *ORSA Journal on computing*, 7, 417-425.
- Johnson, D. S. & McGeoch, L. A. (1995). The Traveling Salesman Problem: A case study, *Local search in Combinatorial Optimization*, E. Aarts and Lenstra, J. K., 215-310, John Wiley and Sons Ltd., John Wiley and Sons Ltd.
- Johnson, D. S. et al. (1996). Asymptotic Experimental Analysis for the Held-Karp Traveling Salesman Bound, *Proceedings of the 7th Annual ACM-SIAM symposium on Discrete Algorithms*, 341-350, January 1996.
- Johnson, D. S. (1990). Local optimization and the traveling salesman problem, *ICALP '90*, 446-461, Springer-Verlag.

- Júnger, M. et al. (1995). The traveling salesman problem, *Network Models, Handbooks in Operations Research and Management Science*, M. O. Ball, Magnanti, T. L., Monma, C. L. and Nemhauser, G. L., 225-330, North-Holland. Amsterdam, North-Holland, Amsterdam.
- Karp, R. M. (1972). Reducibility among Combinatorial Problems, *Complexity of Computer Computations*, R. E. Miller and Thatcher, J. W., 85-103, Plenum Press, New York., Plenum Press, New York.
- Lawler, E. L. et al. (1985). *The Traveling Salesman Problem*, John Wiley and Sons Ltd.
- Lin, S. (1965). Computer solutions of the traveling salesman problem. *The Bell system technical journal*, 44, 2245-2269.
- Lin, S. & Kernighan, B. W. (1973). An effective heuristic algorithm for the traveling salesman problem. *Operations Research*, 21, 498-516.
- Little, J. et al. (1963). An algorithm for the traveling salesman problem. *Operations Research*, 11, 6, 972-989.
- Miliotis, P. (1976). Integer programming approaches to the traveling salesman problem. *Mathematical Programming*, 15, 367-378.
- Miller, C. et al. (1960). Integer programming formulations and traveling salesman problem. *Journal of the Association for Computing Machinery*, 7, 326-329.
- Naddef, D. (1992). The binested inequalities for the symmetric traveling salesman Polytope. *Mathematics of Operations Research*, 17, 4, 882-900.
- Nemhauser, G. L. & Wolsey, L. A. (1988). *Integer and Combinatorial Optimization*, John Wiley and Sons Ltd.
- Osman, I. H. & Kelly, J. P. (1996). *Meta-heuristics : Theory and Applications*, Kluwer Academic Publishers.
- Osman, I. H. (1995). An Introduction to Meta-Heuristics, *Operational Research Tutorial papers*, M. Lawrence and Wilson, C., 92-122, Operational Research Society Press. Birmigham, U.K., Operational Research Society Press, Birmigham, U.K.
- Osman, I. H. (1993). Metastrategy simulated annealing and Tabu search for the vehicle routing problem. *Annals of Operations Research*, 41, 421-432.
- Osman, I. H. (1995). Heuristics for generalized assignment problem, simulated annealing and Tabu search approaches. *OR Spektrum*, 17, 211-228.
- Osman, I. H. & Christofides, N. (1994). Capacitated clustering problems by hybrid simulated annealing and tabu search. *International Transactions in Operational Research*, 1, 317-329.
- Padberg, M. & Rinaldi, G. (1989). A branch-and-cut approach to a traveling salesman problem with side constraints. *Management Science*, 35, 11, 1393-1414.
- Padberg, M. & Rinaldi, G. (1991). A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM review*, 33, 1, 60-100.
- Padberg, M. W. & Hong, S. (1977). On the symmetric travelling salesman problem: A computational study.
- Papadimitriou, C. H. & Steiglitz, K. (1982). *Combinatorial Optimization : Algorithms and Complexity*, Prentice-Hall, Inc.
- Reeves, C. R. (1993). *Modern Heuristic Techniques for Combinatorial Problems*, McGraw-Hill Book Company.
- Rego, C. (1998). Relaxed tours and path ejections for the traveling salesman problem. *European Journal of operational research*, 2, 522-538.

- Reinelt, G. (1992). Fast Heuristics for Large Geometric Traveling Salesman Problems. *ORSA Journal on Computing*, 4, 2, 206-217.
- Reinelt, G. (1994). *The traveling salesman: Computational solutions for TSP applications*, Springer Verlag, 0-387-58334-3.
- Rosenkrantz, D. J. et al. (1977). An analysis of several heuristics for the travelling salesman problem. *Siam Journal of Computing*, 6,3, 1977.
- Rubinshtein, M. I. (1971). On the symmetric TSP. *Automatika I Telemekhanika*, 6, 126-126.
- Sahni, S. & Gonzales, T. (1976). P-complete approximation problem. *J. ACM*, 23, 555-565.
- Sharda, R. (1994). Neural networks for the MS/OR analyst. An application bibliography. *Interfaces*, 24, 116-125.
- Shmoys, D. A. & Williamson, D. P. (1990). Analyzing the Held-Karp TSP bound: A monotonicity property with application. *Information Processing Letters*, 35, 281-285.
- Stewart Jr, W. R. (1987). Accelerated branch exchange heuristics for symmetric travelling salesman problems. *Networks*, 17, 423-437.
- Storer, R. H. et al. (1995). Local search in problem and heuristic space for job shop scheduling. *ORSA Journal on Computing*, 7, 453-458.
- Swamy, M. N. S. & Thulasiraman, K. (1981). *Graphs, Networks, and Algorithms*, John Wiley and Sons Ltd.
- Valenzuela, C. L. & Jones, A. J. (1996). Estimating the Held-Karp lower bound for the geometric TSP. *European Journal of Operational Research*, 102, 157-175.
- Wolsey, L. A. (1980). Heuristic analysis, linear programming and Branch and Bound. *Mathematical Programming Study*, 13, 121-134.

Modelling Access Control with Dynamic Role Binding

Al-Dahoud¹ Ali and Dr.K.Chitra²

¹*Al-Zaytoonah University, Jordan*, ²*Department of Computer Applications, Thiagarajar School of Management Madurai, Tamil Nadu, India*
aldahoud@alzaytoonah.edu.jo, chitra@tsm.ac.in

Abstract

To achieve the goal of realizing object adaptation to environments, we model the object with its role using parameterized UML models. An environment is defined as a field of collaboration between roles and an object adapts to the environment assuming one of the roles. Objects can freely enter or leave environments and belong to multiple environments at a time so that dynamic adaptation or evolution of objects is realized. Organizations use Role-Based Access Control to protect computer-based resources from unauthorized access. This paper describes a method for modeling access control for dynamic role binding using parameterized UML design models. Reusable parameterized UML models are specified as patterns and are expressed using UML template diagrams. Developers can use the models to identify their policy violations. The method is illustrated using a small banking application.

Key Words: Parameterized UML, Object adaptation, Evolution, Reusable.

1. Introduction

Objects represent things or concepts of the real world and it is this representation feature that gives the object-oriented technology the high modeling capability. Objects in the real world exist in various environments. If an object permanently resides in a fixed environment, the structure and behavior of the object can possibly stay unchanged over time. However, environments surrounding objects may not be stable due to various reasons. If objects are humans or manufacturing equipment, their environment changes periodically between the day and the night and between the weekdays and the weekend. When an object moves, the surrounding environment naturally changes. Even if an object stays at the same place for a certain period of time, the environment itself may dynamically change. Corresponding to such environmental change, objects adaptively change themselves. Conversely, objects may spontaneously evolve, causing change in their relation to the environment and that in turn may trigger change in the environment. Moreover, there generally exist multiple environments around an object and the object may selectively

belong to a subset of them at a time and the selection of environments may also change dynamically.

How is such adaptation or evolution of objects handled in the world of object-oriented modeling and programming? As many researchers have pointed out, current widely-used object-oriented modeling and programming languages do not conveniently support such flexibility. Motivation of our research is to build a parameterized UML model that is flexible enough to cope with future changes but simple enough to describe and reason about the design validity.

The model to be described in the succeeding sections has the following features.

1. Objects can freely enter or leave environments and belong to multiple environments at a time so that dynamic adaptation or evolution of objects is realized.
2. Environments and roles are the first class constructs at model description time so that separation of concerns is not only materialized as a static structure but also observed as behaviors.
3. Environments are independent reuse components to be deployed separately from objects that participate in them.

2. Access Control for Dynamic Role Binding (DRBAC)

Roles are considered for listing up functions or behaviors of an object to define a clear boundary of the object, thus their granularity is smaller than objects and conceptually comparable to the level of methods. However, the major motivation of this paper is to design access control for object adaptation to environments using parameterized UML. An environment in the context of role model is regarded as a collaboration field and in order to realize adaptation, objects should be allowed to enter collaboration environments by assuming roles and to leave from environments by discarding roles dynamically.

Dynamic Role Binding Access Control (DRBAC) constraints can be organized as follows: Core DRBAC, Hierarchical DRBAC, Static Separation of Duty Relations, and Dynamic Separation of Duty Relations.

The Core DRBAC requires that users (i.e. Objects) be assigned to environment, users be assigned to roles (job function) corresponding to that environment, roles be associated with permissions (approval to perform an operation on a database), and users acquire permissions by being assigned to roles. The Core DRBAC places a constraint on the cardinalities of the user-role assignment relation that when a user enters an environment he is assigned a role when he/she leaves that environment he/she discards that role. The Core DRBAC does not place any constraint on the cardinalities of the permission-role association. Core DRBAC also includes the notion of user environments. A user enters an environment during which he activates a subset of the roles assigned to him. Each user may belong to multiple environments; however, each environment is associated with only one user. The operations that a user can perform in an environment depend on the roles activated in that environment and the permissions associated with those roles.

Hierarchical DRBAC adds features supporting role hierarchies (RH). Hierarchies are used to describe a structure of roles in an organization. Role hierarchies define an inheritance relation among the roles. Role r1 inherits from role r2 only if all permissions of r2 are also permissions of r1 and all users of r1 are also users of r2. The inheritance relationship is reflexive, transitive and anti-symmetric.

Static Separation of Duty (SSD) relations are necessary to prevent conflict of interests that arise when a user gains permissions associated with conflicting roles (roles that cannot be assigned to the same user). SSD relations are specified for any pair of roles that conflict. The SSD relation places a constraint on the assignment of users to roles, that is, assignment to a role that takes part in an SSD relation prevents the user from being assigned to the related conflicting role. The SSD relationship is symmetric, but it is neither reflexive nor transitive. SSD may exist in the absence of role hierarchies (referred to as SSD DRBAC), or in the presence of role hierarchies (referred to as hierarchical SSD DRBAC). The presence of role hierarchies complicates the enforcement of the SSD relations: before assigning users to roles not only should one check the direct user assignments but also the indirect user assignments that occur due to the presence of the role hierarchies.

Dynamic Separation of Duty (DSD) relations aim to prevent conflict of interests as well. The DSD relations place constraints on the roles that can be activated in a user's environment. If one role that takes part in a DSD relation is activated, the user cannot activate the related (conflicting) role in the same session. A model of DRBAC is shown in Fig. 1.

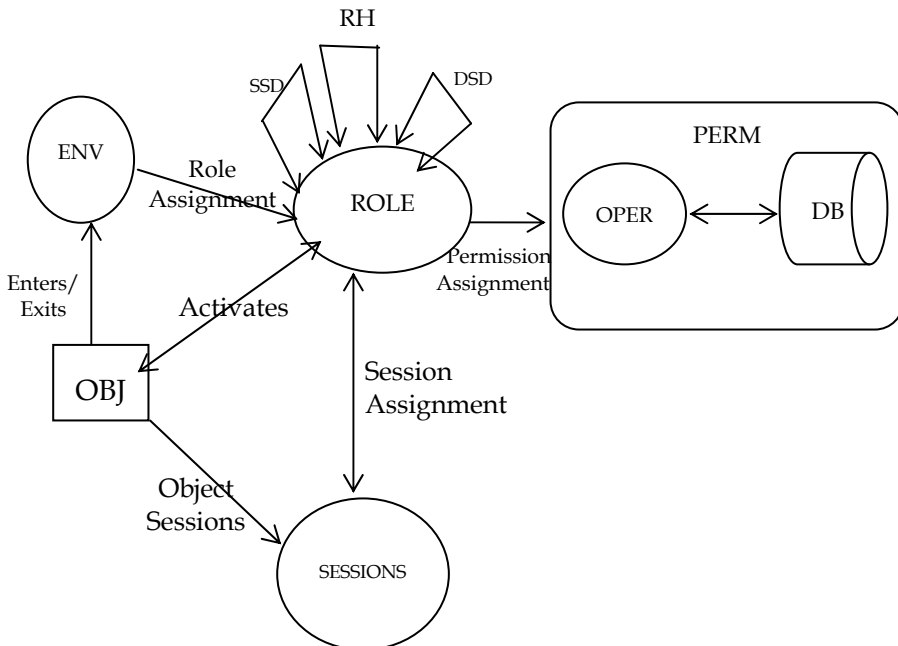


Fig. 1. Access Control in Dynamic Role Binding

The DRBAC in Fig. 1 consists of: 1) a set of objects (OBJ) where an object is an intelligent autonomous agent, 2) a set of environments (ENV) where an object enters or leaves 2) a set

of roles (ROLE) where a role is a job function which is assigned to the object when it enters an environment and is discarded when it exits the environment, 3) the database (DB) where DB is an entity that contains or receives information, 4) a set of operations (OPER) where an operation is an executable image of a program, and 5) a set of permissions (PERM) where a permission is an approval to perform an operation on database. The cardinalities of the relationships are indicated by the absence (denoting one)

or presence of arrows (denoting many) on the corresponding associations.

For example, the association of an object to an environment and the association of an object to role are one-to-many. All other associations shown in the figure are many-to-many. The association labeled Role Hierarchy (RH) defines the inheritance relationship among roles. The association labeled SSD specifies the roles that conflict with each other. The association labeled DSD specifies the roles that cannot be activated within an environment by the same user.

3. A Reusable DRBAC Model

In this section a DRBAC pattern is described as a UML template class diagram. A class diagram is obtained from a template diagram by binding the parameters to values. Fig. 2 shows a class diagram template describing hierarchical DRBAC with SSD and DSD. The symbol “|” is used to indicate parameters.

The class diagram template shown in Fig. 2 consists of class and association templates. A class template is a class descriptor with parameters. Class templates are associated with attribute templates (e.g., |Name : String in Role) and operation templates (e.g., |grantPermission in Role). Association templates (e.g., |UserAssignment) consist of parameters for association names and association-end multiplicities. The OCL constraints in Fig. 2 restrict the values that can be bound to association-end multiplicity parameters. The multiplicity “1” on the UserSessions association-end attached to Object is strict: a session can only be associated with one user.

The User Object class template defines classes that describe Objects. When a user enters an environment, he is assigned a role (assignRole). Then he creates a new session (createSession), delete a session (deleteSession). When he exits that environment, his role is discarded (deassignRole). A UserSessions link (i.e., an instance of an association obtained by binding the parameters of UserSessions to values) is created by a createSession operation (i.e., an operation obtained by binding the operation template parameters to values) and deleted by a deleteSession operation. The operation assignRole creates a RoleAssignment link; the deassignRole removes a RoleAssignment link.

The class template Role is used to produce classes representing roles with behavior that (1) associates a new permission with the role (grantPermission), (2) deletes an existing permission associated with the role (revokePermission), (3) adds an immediate inheriting role (addInheritance), (4) deletes an immediate inheriting role (deleteInheritance), (5) adds a role to the set of conflicting roles (addSSDRole), (6) deletes a role from the existing set of conflicting roles (deleteSSDRole), (7) checks whether the role is in an SSD relationship with a given role in the presence of hierarchies (checkSSD), (8) checks whether the role has a given permission (checkAccess), (9) checks whether the role is in a DSD relation with a given role (checkDSD), (10) deletes a DSD relation between the role and a given role (deleteDSDRole), and (11) adds a DSD relation with a given role (addDSDRole).

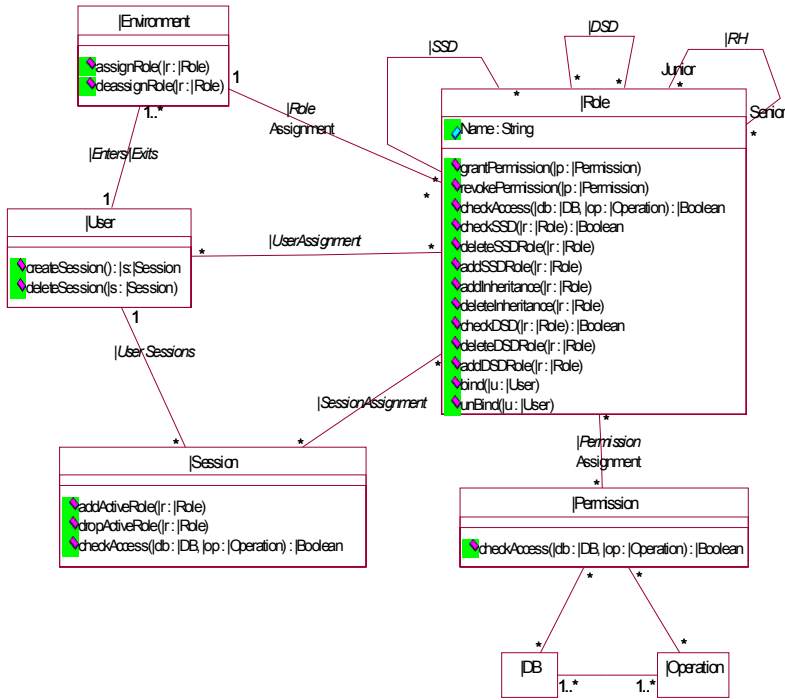


Fig. 2. A DRBAC Class Diagram Template

The class template Session is associated with the template operations: addActiveRole (activates a role in a session), dropActiveRole (deactivates a role in a session), and checkAccess (checks whether the role has the permission to perform an operation on the database).

The class template Permission is associated with an operation template, checkAccess, that checks whether the role has the permission to perform the operation on the database. Each operation template is associated with an OCL template expression that produces OCL pre- and post-conditions when the template parameters are bound to values.

Pre- and post-condition templates associated with the createSession and grantPermission operation templates are given below:

```

context | User:: | createSession():(| s: | Session)
post: result = |s and
|s.oclIsNew() = true and self. | Session → includes(|s)
    
```

```

context | Role:: | grantPermission ( | p: | Permission)
post: self. | Permission includes(| p)
    
```

We express DRBAC constraints that restrict SSD and DSD relationships as OCL template expressions. Examples of these constraints are given below:

- SSD constraint. A user cannot be assigned to two roles that are involved in an SSD relation.

```
context | User inv:
self. | Role → forAll(r1, r2 | r1. | SSD → excludes(r2))
```

- Hierarchical SSD constraint. There cannot be roles in an SSD relation which have the same senior role.

```
context _ Role inv:
let allSenior(r1) = r1.senior -> union(r1.senior -> collect(r2 | allSenior(r2)))
in
self. | SSD -> forAll(r1 | allSenior(r1) -> excludesAll(allSenior(self)))
```

- DSD constraint. A user cannot activate two roles in DSD relation within a session.

```
context | User inv:
| self. | Session. | Activates -> forAll(r1, r2 | r1. | DSD -> excludes(r2))
```

4. DRBAC Model on Banking Application

To illustrate this approach we use a simple banking application taken from [5]. The application is used by various bank officers to perform transactions on customer deposit accounts, customer loan accounts, ledger posting rules, and general ledger reports. The transactions include 1) create, delete, or modify customer deposit accounts, 2) create, delete, or modify customer loan accounts, 3) modify the ledger posting rules, and 4) create general ledger report. When a user enters Bank environment, he is assigned a BANKROLE when he exits bank environment he is assigned another role according to the environment he enters and assigned permissions depending on the role he is assigned. Fig. 3 shows DRBAC class diagram for the bank environment.

5. DRBAC Policies applied using Object Diagrams

DRBAC policies when applied to a role in an environment constrain how system users access system resources. They determine 1) the assignment of roles to system users, 2) the permissions associated with roles in the environment, 3) the inheritance relationships between roles, and 4) the SSD and DSD relationships between roles. In this section we illustrate how DRBAC policies can be described by object diagrams when the user is assigned a ROLE.

The DRBAC model supports the specification of four types of policies: 1) core policies that conform to core DRBAC, that is, policies that determine user-role and role-permission assignments, 2) hierarchical policies that conform to hierarchical DRBAC, that is, policies that determine inheritance relationships between roles, 3) SSD policies that conform to SSD DRBAC, that is, policies that determine what roles are conflicting, and 4) DSD policies that conform to DSD DRBAC, that is, policies that determine what roles to be activated in a session. A set of DRBAC policies for the banking system is given below:

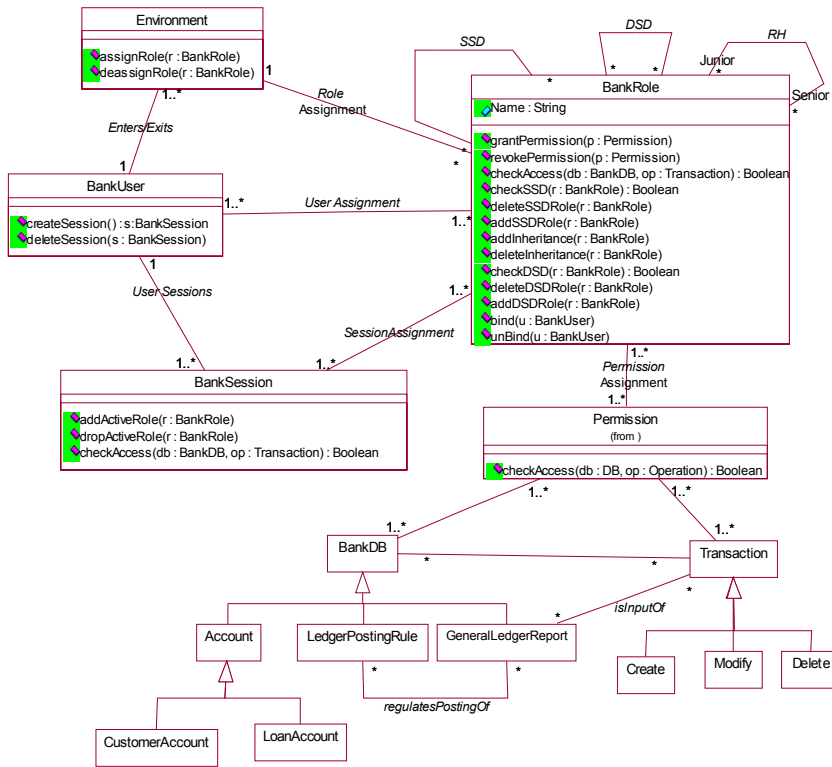


Fig. 3. A DRBAC Class Diagram for a Banking Application

Core policies: The roles of the banking system (instances of BankRole) are teller, customerServiceRep, accountant, accountingManager and loanOfficer. The permissions assigned to these roles are given below:

- P1 A teller can modify customer deposit accounts.
- P2 A customer service representative can create or delete customer deposit accounts.
- P3 An accountant can create general ledger reports.
- P4 An accounting manager can modify ledger-posting rules.
- P5 A loan officer can create and modify loan accounts.

Fig. 4. Shows the object diagrams describing policies P1 to P5 respectively.

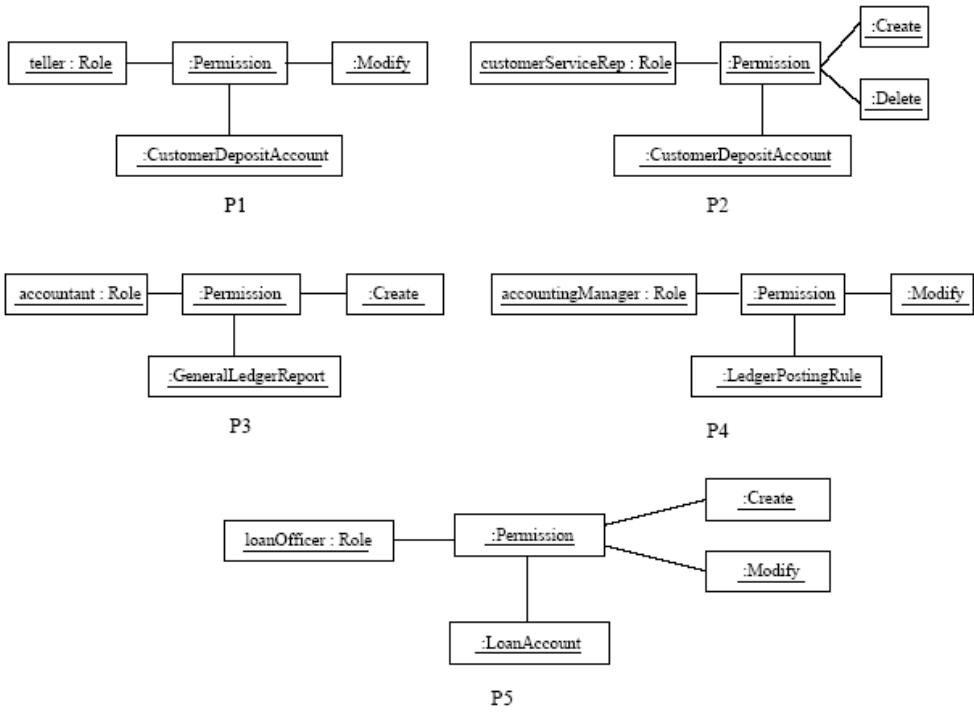


Fig. 4. Object Diagrams for Policies P1 to P5

Hierarchical policies: A role hierarchy defines inheritance relationships between roles. Through the inheritance relationship, a senior role inherits the permissions of its junior roles and any user assigned to the senior role is also assigned to the junior roles. The hierarchical policies in the banking application are stated below:

H1 Customer service representative role is senior to the teller role.

H2 Accounting manager role is senior to the accountant role.

Fig. 5(a),(b) describe policies H1 and H2 respectively.

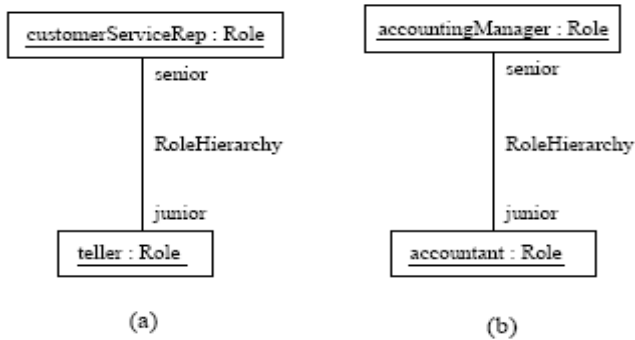


Fig. 5(a), (b) Object Diagrams for Policies H1 and H2

SSD policies: SSD policies prevent a user from being assigned to two conflicting roles. For the banking system the following pairs of roles are conflicting:

- {(teller, accountant), (teller, loanOfficer),
- (loanOfficer, accountant), (loanOfficer, accountingManager),
- (customerServiceRep, accountingManager)}

The object diagram in Fig. 6 describes the SSD RBAC policies.

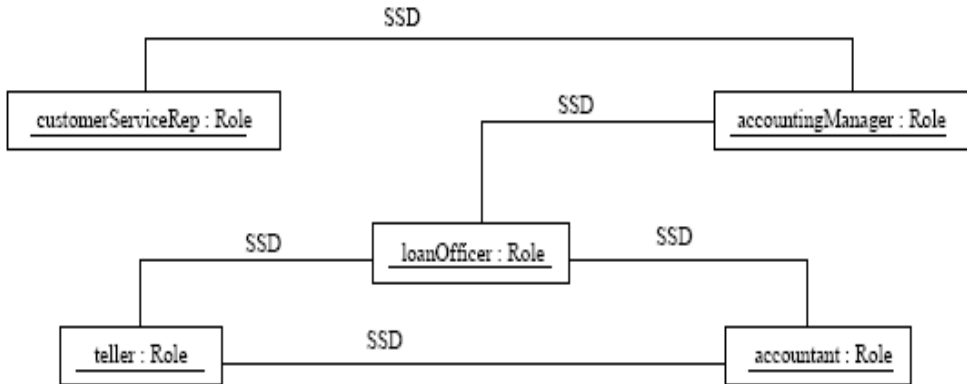


Fig. 6. Object Diagram for SSD Policies

DSD policies: DSD policies prevent a user from playing a role in a session, if another role in a DSD relation has been activated. For the banking system the following pair of roles are in DSD relation:

- {(customerServiceRep, loanOfficer)}
- The object diagram in Fig. 7 describes the DSD DRBAC policy.

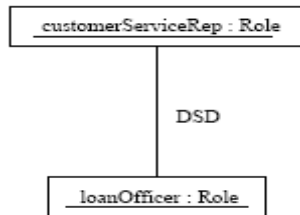


Fig. 7. Object Diagram for DSD Policy

6. Identifying Conflicts in Application-Specific DRBAC Policies

In this section we show how DRBAC violation patterns expressed as object diagram templates can be used to identify conflicts. If a violation pattern exists in an object diagram describing a policy, then a conflict exists. Fig. 8 shows object diagram templates that when instantiated produce object structures that violate DRBAC constraints. Fig. 8(a) describes structures in which a user is assigned to roles in an SSD relationship (violation of the SSD constraint). Fig. 8(b) describes structures in which two roles in an SSD relationship have a

common senior role and structures in which a senior role is in an SSD relationship with a junior role (both are violations of the hierarchical SSD constraint). Fig. 8(c) describes structures in which a user in a session activates two roles that are in a DSD relationship (a violation of the DSD constraint). Formally, an object diagram has the violation described by a violation pattern if there exists a binding that produces an object structure contained in the object diagram.

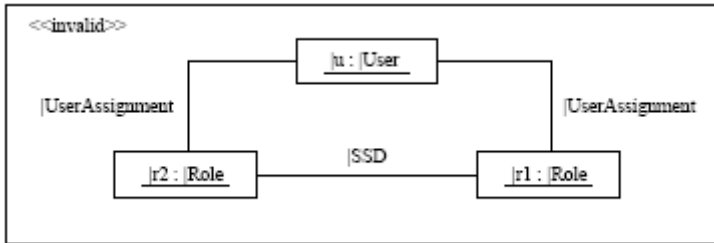


Fig. 8. (a) Violation of SSD Constraint

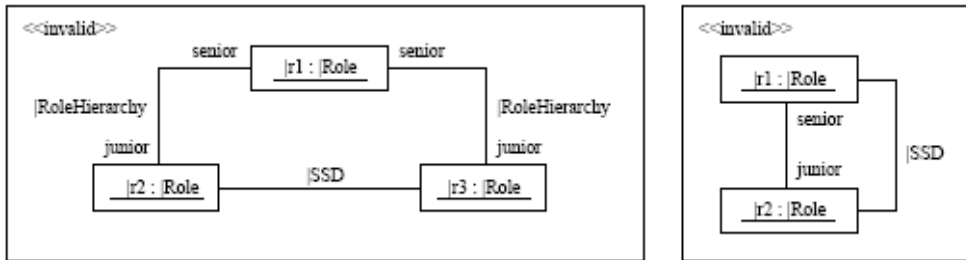


Fig. 8. (b) Violations of Hierarchical SSD Constraint

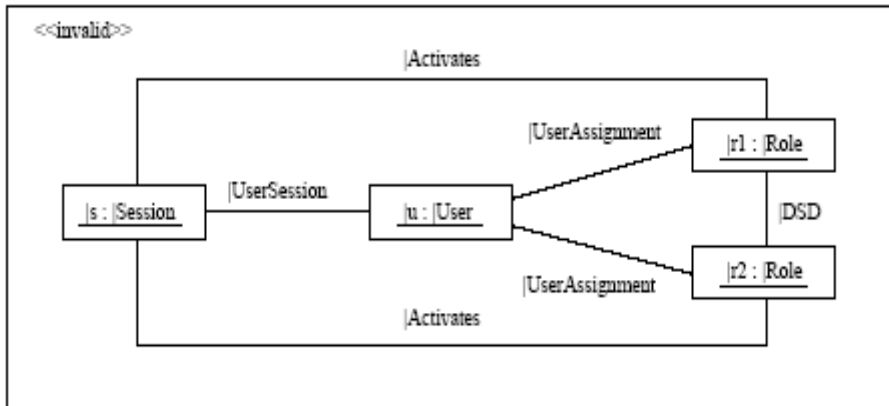


Fig. 8. (c) Violation of DSD Constraint

7. Related Work

Tidswell and Jaeger [21] propose an approach to visualizing access control constraints. They point out the need for visualizing constraints and the limitations of previous work on expressing constraints. A drawback of their work is that they created a new notation for specifying constraints and it is not clear how the new notation can be integrated with other widely-used design notations.

Aspect-oriented programming with AspectJ has a feature of adding aspects dynamically as well as statically [14]. The main objective of writing aspects is to deal with cross-cutting concerns. It implies that there already exists some structure of module decomposition but in adding a new type of concern, related pieces of code are distributed among modules, cross-cutting the existing structure.

Although there have been efforts of designing software from the beginning based on the AOP method under the name of “early aspects”[20], the normal framework of mind for thinking aspects assumes the existing program code as a target of inserting advices to join points.

A large volume of research (e.g., see [2–4, 6, 7, 11, 12, 14]) exists in the area of access control policy specification. Formal logic-based techniques (e.g., see [2–4, 6, 11, 14]) are often used to specify security policies. The use of mathematical concepts and notation that are not familiar to software developers makes them difficult to use and understand. Other researchers have used high-level languages to specify policies [12, 13, 19, 20]. Although high-level languages are easier to understand than formal logic-based approaches, they are not analyzable.

Some work has been done on modeling system security using UML. Jurjens [15] proposes UMLsec, a UML profile for modeling and evaluating security aspects based on the multi-level security model. Lodderstedt et al. propose SecureUML [17], an extension of the UML that defines security concepts. These approaches mainly focus on extending the UML notation to better reflect security concerns.

8. Conclusion

The work described in this paper focuses on specifying the dynamic role binding and access control when a role is assigned to an Object. Checking for the presence of a pattern in an object diagram specifying a set of policies is essentially a search for a sub graph in an object diagram. The approach of binding objects and roles have the following characteristics: Composition takes place when an object instance and a role instance are bound together; an object instance can be bound to multiple role instances residing in different environments; when an object enters an environment it is assigned a role, when it exits that environment the associated role is discarded by the object.

9. References

1. G.J. Ahn and R. Sandhu. Role-based Authorization Constraints Specification. *ACM Transactions on Information and Systems Security*, 3(4):207–226, November 2000.
2. S. Barker. Security Policy Specification in Logic. In *Proceedings of the International Conference on Artificial Intelligence*, pages 143–148, Las Vegas, NV, 2000.

3. S. Barker and A. Rosenthal. Flexible Security Policies in SQL. In Proceedings of the 15th Annual IFIP WG 11.3 Working Conference on Data and Applications Security, Niagara-on-the-Lake, Canada, 2001.
4. E. Bertino, P. Bonatti, and E. Ferrari. TRBAC: A Temporal Role-Based Access Control Model. In Proceedings of the 5th ACM Workshop on Role-Based Access Control, pages 21–30, Berlin, Germany, 2000.
5. R. Chandramouli. Application of XML Tools for Enterprise-Wide RBAC Implementation Tasks. In Proceedings of 5th ACM workshop on Role-based Access Control, Berlin, Germany, July 2000.
6. F. Chen and R. Sandhu. Constraints for Role-Based Access Control. In Proceedings of the 1st ACM Workshop on Role-Based Access Control, Gaithersburg, MD, 1995.
7. N. Damianou and N. Dulay. The Ponder Policy Specification Language. In Proceedings of the Policy Workshop, Bristol, U.K., 2001.
8. D.F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli. Proposed NIST Standard for Role-Based Access Control. *ACM Transactions on Information and Systems Security*, 4(3), August 2001.
9. Geri Georg, Robert France, and Indrakshi Ray. An Aspect-Based Approach to Modeling Security Concerns. In Proceedings of the Workshop on Critical Systems Development with UML, Dresden, Germany, 2002.
10. Geri Georg, Indrakshi Ray, and Robert France. Using Aspects to Design a Secure System. In Proceedings of the International Conference on Engineering Complex Computing Systems (ICECCS 2002), Greenbelt, MD, December 2002. ACM Press.
11. R. J. Hayton, J. M. Bacon, and K. Moody. Access Control in Open Distributed Environment. In IEEE Symposium on Security and Privacy, pages 3–14, Oakland, CA, May 1998.
12. M. Hitchens and V. Varadarajan. Tower: A Language for Role-Based Access Control. In Proceedings of the Policy Workshop, Bristol, U.K., 2001.
13. J. A. Hoagland, R. Pandey, and K. N. Levitt. Security Policy Specification Using a Graphical Approach. Technical Report CSE-98-3, Computer Science Department, University of California Davis, July 1998.
14. S. Jajodia, P. Samarati, and V. S. Subrahmanian. A Logical Language for Expressing Authorizations. In IEEE Symposium on Security and Privacy, pages 31–42, Oakland, CA, May 1997.
15. J. Jurjens. UMLsec: Extending UML for Secure Systems Development. In Proceedings of Fifth International Conference on the Unified Modeling Language, pp. 412–425, pages 412–425, Dresden, Germany, October 2002.
16. Dae-Kyoo Kim, Robert France, Sudipto Ghosh, and Eunjee Song. Using Role-Based Modeling Language (RBML) as Precise Characterizations of Model Families. In Proceedings of the International Conference on Engineering Complex Computing Systems (ICECCS 2002), Greenbelt, MD, December 2002. ACM Press.
17. T. Lodderstedt, D. A. Basin, and J. Doser. SecureUML: A UML-Based Modeling Language for Model-Driven Security. In Proceedings of Fifth International Conference on the Unified Modeling Language, pages 426–441, Dresden, Germany, October 2002.
18. B.T. Messmer and H. Bunke. Subgraph Isomorphism in Polynomial Time. In Lecture Notes in Computer Science Graph Theory - ECCV'98, Springer-Verlag, Berlin, 1998.

19. OASIS. XACML Language Proposal, Version 0.8. Technical report, Organization for the Advancement of Structured Information Standards, January 2002. Available electronically from <http://www.oasis-open.org/committees/xacml>.
20. C. Ribeiro, A. Zuquete, and P. Ferreira. SPL: An Access Control Language for Security Policies with Complex Constraints. In Proceedings of the Network and Distributed System Security Symposium, San Diego, CA, February 2001.
21. J. E. Tidswell and T. Jaeger. An Access Control Model for Simplifying Constraint Expression. In Proceedings of 7th ACM conference on Computer and communications security, pages 154–163, Athens, Greece, November 2000.
22. G. B. and William Cook. Mixin-based inheritance. In OOPSLA 1990, pages 303–311, 1990.
23. D. Bardou and C. Dony. Split objects: a disciplined use of delegation within objects. In OOPSLA '96, pages 122–137, San Jose, California, USA, Oct. 1996.
24. K. Beck and W. Cunningham. A laboratory for teaching object-oriented thinking. In OOPSLA '89, pages 1–6, 1989.

On the Accuracy of a Stewart Platform: Modelling and Experimental Validation

Mircea Neagoe¹, Dorin Diaconescu¹, Codruta Jaliu¹, Sergiu-Dan Stan²,
Nadia Cretescu¹ and Radu Saulescu¹

¹ *Transilvania University of Brasov, Romania*

² *Technical University of Cluj-Napoca, Romania*

1. Introduction

The accuracy modelling of parallel-type robot structures is based on finding out the position and orientation errors of the end-effector in relation with the modelled error-sources. In the case of small (infinitesimal) deviations, the error model is expressed through a *linear function* using special matrices called *error Jacobians* (Neagoe, 2001). Hence, the main objective of accuracy modelling is to express the *error Jacobians*.

The accuracy modelling of the parallel robots represents a real challenge for the researchers, due to the high level difficulties implied in the direct modelling of the errors and, as a result, in the error-Jacobian description. Unlike the case of the serial robots, the analytical expression of the direct error-Jacobian for parallel structures is generally inaccessible or very complex; contrary, the inverse Jacobian can be obtained without major difficulties.

The studies referring to the precision modelling of parallel structures are scarcely found in literature, the most of the papers dealing especially with the inverse kinematics problem of parallel structures. There can not be found important results concerning the direct Jacobian modelling, neither a generalization of modelling, due to the severe difficulties of modelling. Many papers present specific solutions for different specific parallel structures: Tau parallel robot (Cui et al., 2008), six-dof parallel kinematic machine Linapod (Pott & Hiller, 2008), a 4-DOF parallel manipulator H4 (Wu & Yin, 2008), 3-DOF planar parallel robots (Briot and Bonev, 2008), etc. Currently, the problem of the parallel robots kinematics is reduced to the expression of the inverse kinematic Jacobian.

Three representative methods applied in kinematic modelling of parallel robots were identified:

1. *The partial derivatives method*, which consists firstly in identifying the geometric relations for the modelled parallel structure and, than, the partial derivation relative to the independent parameters (Merlet, 1990; Merlet & Gosselin, 1991). Generally, the obtained model can be expressed by the relation $\mathbf{A} \cdot d\mathbf{q} = \mathbf{B} \cdot d\mathbf{X}$, used for the joint velocities $d\mathbf{q}/dt$ calculus in relation with the operational velocities $d\mathbf{X}/dt$. In this case, the inverse matrix \mathbf{A}^{-1} is easily obtained, \mathbf{A} being a square matrix for Stewart platforms. On the other hand, to express

the operational deviations $d\mathbf{X}$ and, implicitly, the direct Jacobean assumes to inverse the matrix \mathbf{B} , which has the dimension equal to the number of independent modelling parameters.

2. *The vectorial method*, which expresses the articular velocities through the vectorial relations of the kinematic modelling of velocities (Benea, 1996; Merlet, 1990).
3. *The kinematic screws method*, which obtains the kinematic model by vectorial transformations applied to the plückerien coordinates of a line in space (Ficher, 1986; Lee et al., 1999; Toyama & Hatae, 1989).

Starting from the previous revealed aspects, the authors propose in the paper a general method, based on the use of homogenous operators, useful for accuracy modelling of parallel structures with any configuration and complexity, with application for a Stewart-Deltalab parallel platform.

The chapter is structured in four main sections. The first section introduces the theoretical background on the proposed method meant for accuracy modelling of parallel robots and presents the steps and mathematical support of a general algorithm derived from this method.

In the second part, the proposed modelling is concretely applied to derive the accuracy model of the Stewart-Deltalab platform, considering the independent kinematic parameters (independent joint variables) as error-sources. Numerical examples will be presented, based on the analytical closed-form accuracy model previously obtained.

The third section will contains the authors' contribution to the modelling of the Stewart-Deltalab platform accuracy, applying the same proposed modelling method and algorithm, considering a set of geometric parameters as error-sources. Also, numerical examples will be done.

In the last section, relevant aspects regarding the experimental testing of the error models used in accuracy studies of spatial parallel-structures will be presented. The theoretical accuracy models of the Stewart-Deltalab platform are verified by experimental testing, in conformity with a concrete experimental research program and a specific mathematical support.

2. Theoretical background

The method proposed for parallel robot accuracy modelling includes three main steps:

1. *Breaking* of parallel structure into *open kinematic chains* (OKC) and description of *the error models* for the obtained OKC, *considered as independent chains*, by applying the specific error modelling of serial robots (Gogu, 1995; Gogu et al., 1997).
2. *Recovering* the parallel structure by assembling the error models of OKC and finally expressing the *dependent errors*.
3. *Description of the end-effector errors* related to the independent errors, by replacing the dependent errors in the error model derived for one of the OKC (step 1) with their corresponding expressions (identified at step 2).

Explanatory notes:

— Structurally, a parallel robot includes:

- *Active (actuated) joints*; the relative displacements in the actuated joints are the *robot generalized variables (independent joint variable)* of the structure. Thus, the deviations

of these variables become input errors (*independent errors*) in the accuracy modelling.

- *Passive (non-actuated) joints*; these joints are included in any parallel structure in order to obtain parallel-type links. The relative displacements in passive joints are functions of independent joint variables and, hence, displacement errors in passive joints, called *dependent errors*, depend on the independent errors.
- Modelling the influence of geometrical parameters on the end-effector accuracy is done on an *equivalent structure*, obtained from the initial structure by associating *fictive joints* to the geometrical parameters affected by errors: a prismatic joint is introduced for each linear error and a revolute joint – for each angular error.
 - The proposed modelling requires only the inverse geometrical model of the parallel structure and the direct geometrical models of OKC. All these models are expressed, generally, without significant difficulty.

The general case of a parallel robot structure (Fig. 1,a) is considered; it consists of a mobile platform (m) connected to the fixed one (f) by p kinematic chains (legs) $A_i B_i$, in a parallel layout. The spatial guiding of the mobile platform is obtained by actuating n joints nominated as *active* joints. In practice, generally $p = n$, each leg including only one active joint. For the sake of clarity, the following *explanations* have to be mentioned:

- For the simplicity, but without reducing the generality, Figure 1 shows only symmetrical legs in a parallel layout and their connecting joints to the base and moving platforms. In a general case, the robot legs can have different structures and usually include intermediate joints.
- The breakage of the parallel structure can be done in different ways: to the characteristic point of the end-effector, to the base joints or to any intermediate joints.

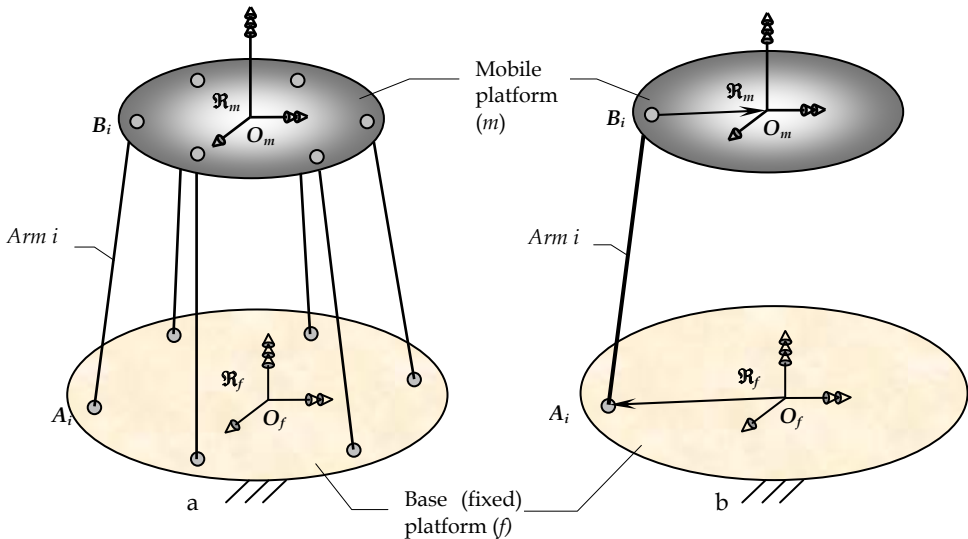


Fig. 1. Parallel robot structure in a general configuration (a), and the open chain i obtained by "disassembling" the parallel structure (b)

- In the following modelling, the independent parameters will be designated by notation \mathbf{p}_{ind} while the dependent parameters by \mathbf{p}_{dep} ; \mathbf{p}_{ind}^k and \mathbf{p}_{dep}^k designate independent parameters and respectively dependent parameters for the leg k (OKC k) of the parallel structure.
- The independent parameters \mathbf{p}_{ind} designate *active joint variables*, in the case of joint error modelling, or *geometrical parameters* when the error model for geometrical deviations has to be established. In the global error modelling, both types of kinematic and geometrical parameters are considered independent.
- The end-effector errors can be reduced in the base, end-effector or other intermediate reference frame. In order to simplify the notations, the reference frame where the errors are reduced is not specified.

The proposed modelling begins with a preliminary step: *breaking* the parallel structure into open kinematic chains (OKC); there are many splitting variants that can be applied, each of them giving specific features to the modelling algorithm (Neagoe, 2001). The variant used in the paper is based on breaking the parallel structure in the origin O_m of the mobile frame \mathcal{R}_m , obtaining the open chains $O_f A_i B_i O_m$, $i = 1..p$ (Fig. 1,b); all the open chains have the mobile platform (end-effector) as the final element. The obtained p independent OKC have the same property: *their extremity points are permanently coincident* and, consequently, the end-effector's errors for all the p OKC are *identically!*

Further on, the steps of the proposed modelling algorithm and its specific mathematical aspects are briefly presented.

Step I: deriving the end-effector errors for the p OKC

By applying the well known relations for open kinematic chains accuracy modelling (Gogu, 1995; Gogu et al., 1997 ; Paul, 1981), in the case of OKC i will be obtained:

$$\begin{bmatrix} d_x \\ d_y \\ d_z \\ \delta_x \\ \delta_y \\ \delta_z \end{bmatrix} = [\mathbf{J}_{ind}^i] [d\mathbf{p}_{ind}^i] + [\mathbf{J}_{dep}^i] [d\mathbf{p}_{dep}^i], \quad (1)$$

where \mathbf{J}_{ind}^i is the error Jacobean for the independent errors and \mathbf{J}_{dep}^i - the error Jacobean for the dependent errors of the parallel structure, $i = 1..p$. In this step, having only independent open kinematic chain (OKC), all the modelling parameters are characterized by independent errors.

Step II: identification of the dependent errors

In the parallel structure, the end-effector's errors are the same for all the p OKC (*the existence condition of a parallel structure*). As a result, the following $p-1$ independent matrix equations are obtained:

$$[\mathbf{J}_{ind}^j] [d\mathbf{p}_{ind}^j] + [\mathbf{J}_{dep}^j] [d\mathbf{p}_{dep}^j] = [\mathbf{J}_{ind}^k] [d\mathbf{p}_{ind}^k] + [\mathbf{J}_{dep}^k] [d\mathbf{p}_{dep}^k], \quad j \neq k. \quad (2)$$

Without reducing generality, the following assumptions can be accepted: $j = 1$ and $k = 2..p$ or $j = 1..p-1$ and $k = j+1$. Grouping together only the dependent terms from the left member of the equation (2), the relation (3) will be obtained:

$$[\mathbf{J}_{dep}^1][d\mathbf{p}_{dep}^1] - [\mathbf{J}_{dep}^k][d\mathbf{p}_{dep}^k] = [\mathbf{J}_{ind}^k][d\mathbf{p}_{ind}^k] - [\mathbf{J}_{ind}^1][d\mathbf{p}_{ind}^1]. \quad (3)$$

The $p-1$ equations (3) are grouped into a matrix system (4):

$$[\mathbf{J}_{dep}][d\mathbf{p}_{dep}] = [\mathbf{J}_{ind}][d\mathbf{p}_{ind}]. \quad (4)$$

where:

$$[\mathbf{J}_{dep}] = \begin{bmatrix} \mathbf{J}_{dep}^1 & -\mathbf{J}_{dep}^2 & 0 & \cdots & 0 \\ \mathbf{J}_{dep}^1 & 0 & -\mathbf{J}_{dep}^3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & 0 \\ \mathbf{J}_{dep}^1 & 0 & 0 & \cdots & -\mathbf{J}_{dep}^n \end{bmatrix}, \quad [\mathbf{J}_{ind}] = \begin{bmatrix} -\mathbf{J}_{ind}^1 & \mathbf{J}_{ind}^2 & 0 & \cdots & 0 \\ -\mathbf{J}_{ind}^1 & 0 & \mathbf{J}_{ind}^3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & 0 \\ -\mathbf{J}_{ind}^1 & 0 & 0 & \cdots & \mathbf{J}_{ind}^n \end{bmatrix},$$

$$[d\mathbf{p}_{dep}] = [d\mathbf{p}_{dep}^1 \quad d\mathbf{p}_{dep}^2 \quad \cdots \quad d\mathbf{p}_{dep}^p], \quad [d\mathbf{p}_{ind}] = [d\mathbf{p}_{ind}^1 \quad d\mathbf{p}_{ind}^2 \quad \cdots \quad d\mathbf{p}_{ind}^p].$$

Finally, the dependent errors can be expressed through the following relation:

$$[d\mathbf{p}_{dep}] = [\mathbf{J}_{dep}]^{-1} \cdot [\mathbf{J}_{ind}] \cdot [d\mathbf{p}_{ind}] = [\mathbf{J}_{dep}^*][d\mathbf{p}_{ind}]. \quad (5)$$

The central problem of this modelling is to reverse the matrix \mathbf{J}_{dep} . For kinematically determinate structures, matrix \mathbf{J}_{dep} is always a square matrix of $s \times s$ dimension; s is equal to the number of dependent parameters and does not depend on the number of independent parameters considered in modelling. For the structures with a reduced complexity, the reversion can be obtained analytically; for the other cases, a numerical approach is recommended.

Step III: end-effector errors establishment

The end-effector errors can be expressed by introducing the dependent error expressions (rel. 5) into (rel. 1), particularised for each arm. Considering the chain i , the end-effector errors become:

$$\begin{bmatrix} d_x \\ d_y \\ d_z \\ \delta_x \\ \delta_y \\ \delta_z \end{bmatrix}^i = [\mathbf{r}_{ind}^i][d\mathbf{p}_{ind}^i] + [\mathbf{J}_{dep}^i][\mathbf{r}_{dep}^{i*}][d\mathbf{p}_{ind}^i] = [\mathbf{J}^i][d\mathbf{p}_{ind}^i]. \quad (6)$$

Introducing the notation

$$[\mathbf{J}_{dep}^i][\mathbf{r}_{dep}^{i*}] = [\mathbf{J}^i] = [\mathbf{J}_1^i \quad \mathbf{J}_2^i \quad \cdots \quad \mathbf{J}_i^i \quad \cdots \quad \mathbf{J}_p^i], \quad (7)$$

the error-Jacobian \mathbf{J} of the parallel structure can be described as:

$$[\mathbf{J}] = [\mathbf{J}_1^* \ \mathbf{J}_2^* \ \cdots \ \mathbf{J}_i^* + \mathbf{J}_{ind}^i \ \cdots \ \mathbf{J}_p^*]. \quad (8)$$

The Jacobean matrix \mathbf{J} given by relation (8) describes the linear transformation of independent errors into operational errors associated to the end-effector. The complexity of the Jacobean \mathbf{J} depends on the reference frame used for reducing the errors. Most frequently in practice, the Jacobean \mathbf{J} is reduced in the final reference frame \mathfrak{R}_m or in the fixed frame \mathfrak{R}_f (Fig. 1).

3. The Stewart platform presentation

The Stewart DELTALAB platform (Fig. 2) is a parallel manipulator, composed by a moving platform connected to the base through 6 telescopic legs (of variable length). The links between the six legs and the two platforms are materialized by spherical joints.

The parallel structure geometry is completely defined by the coordinates of the points A_i și B_i (the centres of spherical joints, Fig. 3), which can be established by means of parameters $r_f = 270$ mm; $r_m = 195$ mm; $\alpha = 4.25^\circ$; $\beta = 5.885^\circ$ (Fig. 2).

As a result, the analyzed Stewart platform can be geometrically defined as follows (Fig. 2):

The fixed platform:

- The fixed coordinate system attached to the fixed platform: $\mathfrak{R}_f(O_f x_f y_f z_f)$, is placed in the plate's centre (the centre of the circle of radius r_f).
- The centres of the spherical joints, formed of the six cylinders (legs) with the fixed platform, are placed in points A_i , distributed on the circle of radius r_f .
- The points A_i are organized in equidistant groups formed of two appropriate adjacent points separated by the angle 2α .

The moving platform:

- The mobile coordinate system: $\mathfrak{R}_m(O_m x_m y_m z_m)$, placed in the plate's centre (the centre of the circle of radius r_m).
- The centre points B_i of the spherical joints, distributed on a circle of radius r_m .
- The points B_i are also organized in equidistant groups with the centre angle 2β .

The platform initial position (at minimum high) is characterized through the position of point $O_{m'}$ in \mathfrak{R}_f (see Fig. 2), the moving platform being parallel to the base. The distance between $O_{m'}$ and O_f in this position is defined by the parameter $h = O_f O_{m'} = 326.679$ mm, for which the minimum length of the cylinders (active joints) is $L_i = A_i B_i = 387$ mm.

The spatial guidance of the mobile platform related to the fixed one, is described by a set of 6 parameters:

- 3 position (displacement) parameters, given by the coordinates of the point O_m in relation to the reference frame $\mathfrak{R}_{m'}$ attached to the moving platform, expressed in the fixed reference frame \mathfrak{R}_f ,

$$\overrightarrow{O_{m'} O_m} = x_m \vec{i}_f + y_m \vec{j}_f + z_m \vec{k}_f. \quad (9)$$

- 3 orienting parameters: the angles θ_1 , θ_2 și θ_3 , which characterize the reference frame \mathfrak{R}_m orienting in relation to the reference frame $\mathfrak{R}_{m'}$; the associated rotational matrix is described by the following relation:

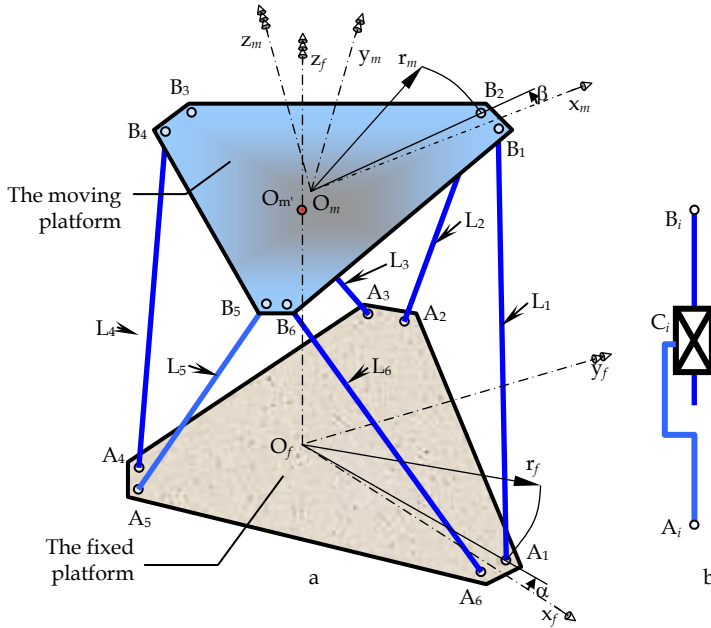


Fig. 2. The structure of the Stewart-DELTALAB platform (a) and of the leg i (b)

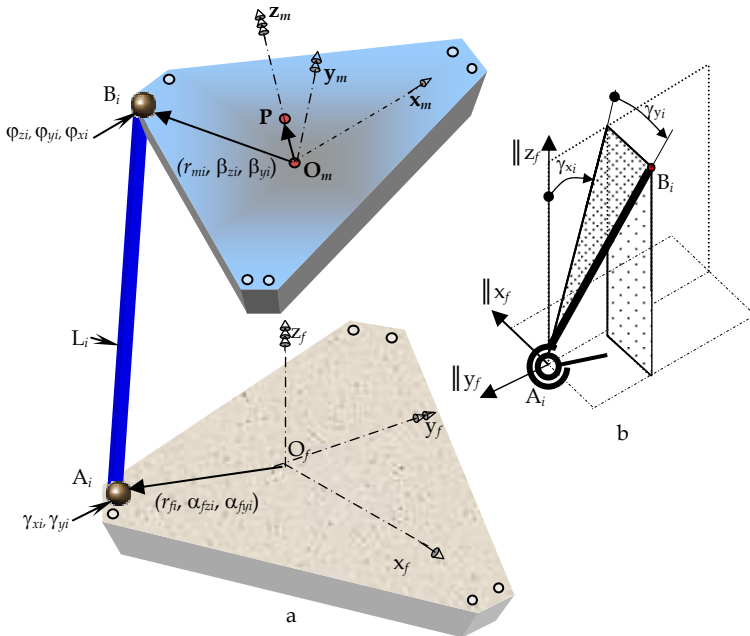


Fig. 3. Decomposition of the parallel structure in open chains and their parameterization

$$R_{m'm} = R_z(\theta_1) \cdot R_x(\theta_2) \cdot R_y(\theta_3) = \begin{bmatrix} c\theta_1 c\theta_3 - s\theta_1 s\theta_2 s\theta_3 & -s\theta_1 c\theta_2 & c\theta_1 s\theta_3 + s\theta_1 s\theta_2 c\theta_3 \\ s\theta_1 c\theta_3 + c\theta_1 s\theta_2 s\theta_3 & c\theta_1 c\theta_2 & s\theta_1 s\theta_3 - c\theta_1 s\theta_2 c\theta_3 \\ -c\theta_2 s\theta_3 & s\theta_2 & c\theta_2 c\theta_3 \end{bmatrix}, \quad (10)$$

where $c\theta_i = \cos(\theta_i)$ and $s\theta_i = \sin(\theta_i)$.

The position of the points A_i in the fixed reference frame \mathfrak{R}_f is defined through a set of spherical coordinates $(\alpha_{zfi}, \alpha_{yfi}, r_{fi})$, and of the points B_i through coordinates $(\beta_{zmi}, \beta_{ymi}, r_{mi})$ defined in \mathfrak{R}_m (Fig. 3). The relative angular displacements from joints A_i are modelled by means of angles γ_{xi} and γ_{yi} (Fig. 3,b) and the displacements from the spherical joints B_i by means of angles $(\varphi_{zir}, \varphi_{yir}, \varphi_{xi})$.

4. Direct error modelling

The objective of the *direct error modelling* is to establish the operational errors of the end-effector related to the values of the source errors; in the paper both active joint variable errors and geometric errors are considered.

4.1 Joint variable errors modelling

In this section, the error sources considered in the accuracy modelling are nominated by the *relative displacements* from the active joints (the prismatic joints C_i , Fig. 2,b). In the infinitesimal errors hypothesis, the error model becomes a *linear model*, where an *error Jacobean* \mathbf{J}_L describes the influence of the independent kinematical parameters (L_1, \dots, L_6) on the end-effector accuracy. This modelling is based on the following assumptions: a) the modelling of I order errors is used; b) the geometrical parameters of both moving and fixed platforms have no errors (ideal geometry); c) the *passive joints* (joints A_i and B_i) are ideal joints; d) the error model expresses the end-effector errors for the characteristic point P (Fig. 3,a); e) The errors are reduced in the end-effector reference frame \mathfrak{R}_p .

In these assumptions, the linear error model is expressed by the following relation:

$$[d\mathbf{X}]_p = [\mathbf{J}_L]_p \cdot [d\mathbf{L}], \quad (11)$$

where $[d\mathbf{X}]_p = [d_x \ d_y \ d_z \ \delta_x \ \delta_y \ \delta_z]_p^T$ is the 6 dimensions vector of the operational errors of the end-effector, for point P , reduced in \mathfrak{R}_p , while $[d\mathbf{L}] = [dL_1 \ dL_2 \ dL_3 \ dL_4 \ dL_5 \ dL_6]^T$ is the vector of the active joint variable errors.

According to the error model (rel. 11), the *central objective* of this modelling is to establish the *error Jacobean* \mathbf{J}_L (through a similar approach, the error Jacobean \mathbf{J}_L can be expressed also in any other frame).

In order to describe the Jacobean \mathbf{J}_L , the former algorithm (section 2) is proposed further on.

Step 1. Description of the end-effector errors for the six open chains (OKC)

Due to the fact that the effector errors are described in the reference frame \mathfrak{R}_p , the direct kinematic modelling for finite displacements of OKC must be done with the homogenous operators of *D-F type (type K)* (Gogu, 1995; Gogu et al., 1997). The kinematic model will also include, in this step, the relative displacements from the passive joints A_i and B_i as *independent parameters*.

The operational errors of each OKC can be now described easily by applying the well-known relations for open chains (Gogu, 1995; Gogu et al., 1997):

$$\begin{bmatrix} d_{xi} \\ d_{yi} \\ d_{zi} \\ \delta_{xi} \\ \delta_{yi} \\ \delta_{zi} \end{bmatrix}_p^i = [\mathbf{J}_i] \begin{bmatrix} \delta\gamma_{xi} \\ \delta\gamma_{yi} \\ dL_i \\ \delta\varphi_{zi} \\ \delta\varphi_{yi} \\ \delta\varphi_{xi} \end{bmatrix}, \quad [\mathbf{J}_i] = \begin{bmatrix} J_{\gamma_{xi}} & J_{\gamma_{yi}} & J_{L_i} & J_{\varphi_{zi}} & J_{\varphi_{yi}} & J_{\varphi_{xi}} \end{bmatrix}, \quad i = 1..6, \quad (12)$$

where the column vectors from matrix \mathbf{J}_i describe the influence of the errors of the modelling parameters.

Step 2. Identification of dependent errors

By splitting the parallel structure into open chains, the passive joints A_i and B_i became fictively actuated and, so, the displacements errors from these joints became independent too. In the parallel structure, all these errors are *dependent* of the independent error sources L_i . That's why, to express the dependent errors related to the independent ones represents the objective of this step; this desideratum becomes possible by modelling the recovering of the parallel connections of the Stewart platform, for which the following condition is used: *the effector errors are the same for all the six OKC*; analytically, the condition is expressed through the following equalities:

$$\begin{bmatrix} d_{xi} \\ d_{yi} \\ d_{zi} \\ \delta_{xi} \\ \delta_{yi} \\ \delta_{zi} \end{bmatrix}_p = [\mathbf{J}_1] \begin{bmatrix} \delta\gamma_{x1} \\ \delta\gamma_{y1} \\ dL_1 \\ \delta\varphi_{z1} \\ \delta\varphi_{y1} \\ \delta\varphi_{x1} \end{bmatrix} = \dots = [\mathbf{J}_6] \begin{bmatrix} \delta\gamma_{x6} \\ \delta\gamma_{y6} \\ dL_6 \\ \delta\varphi_{z6} \\ \delta\varphi_{y6} \\ \delta\varphi_{x6} \end{bmatrix}, \quad (13)$$

which lead to 5 independent matrix equations. Applying separation of dependent terms from the independent ones, the following relation is obtained:

$$\begin{bmatrix} J_{\gamma_{x1}} & J_{\gamma_{y1}} & J_{\varphi_{z1}} & J_{\varphi_{y1}} & J_{\varphi_{x1}} \end{bmatrix} \begin{bmatrix} \delta\gamma_{x1} \\ \delta\gamma_{y1} \\ \delta\varphi_{z1} \\ \delta\varphi_{y1} \\ \delta\varphi_{x1} \end{bmatrix} - \begin{bmatrix} J_{\gamma_{xk}} & J_{\gamma_{yk}} & J_{\varphi_{zk}} & J_{\varphi_{yk}} & J_{\varphi_{xk}} \end{bmatrix} \begin{bmatrix} \delta\gamma_{xk} \\ \delta\gamma_{yk} \\ \delta\varphi_{zk} \\ \delta\varphi_{yk} \\ \delta\varphi_{xk} \end{bmatrix} = J_{L_k} dL_k - J_{L_1} dL_1, \quad (14)$$

for $k = 2..6$.

The systems (14) are assembled into one matrix equation:

$$[\mathbf{J}_\Phi^*] \cdot [\delta\Phi] = [\mathbf{J}_L^*] \cdot [dL], \quad (15)$$

where $[\delta\Phi] = [\delta\gamma_{x1} \ \delta\gamma_{x2} \ \delta\varphi_{z1} \ \delta\varphi_{y1} \ \delta\varphi_{x1} \ \dots \ \delta\varphi_{y6} \ \delta\varphi_{x6}]^T$ is the global vector of dependent errors, while

$$\begin{aligned}
[\mathbf{J}_\Phi^*] &= \begin{bmatrix} \mathbf{J}_{\Phi 1} & -\mathbf{J}_{\Phi 2} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{J}_{\Phi 1} & \mathbf{0} & -\mathbf{J}_{\Phi 3} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{J}_{\Phi 1} & \mathbf{0} & \mathbf{0} & -\mathbf{J}_{\Phi 4} & \mathbf{0} & \mathbf{0} \\ \mathbf{J}_{\Phi 1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{J}_{\Phi 5} & \mathbf{0} \\ \mathbf{J}_{\Phi 1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{J}_{\Phi 6} \end{bmatrix}, \quad [\mathbf{J}_L^*] = \begin{bmatrix} -J_{L1} & J_{L2} & 0 & 0 & 0 & 0 \\ -J_{L1} & 0 & J_{L3} & 0 & 0 & 0 \\ -J_{L1} & 0 & 0 & J_{L4} & 0 & 0 \\ -J_{L1} & 0 & 0 & 0 & J_{L5} & 0 \\ -J_{L1} & 0 & 0 & 0 & 0 & J_{L6} \end{bmatrix}, \quad (16) \\
\mathbf{J}_{\Phi k} &= [J_{\gamma_{ik}} \quad J_{\gamma_{jk}} \quad J_{\varphi_{ik}} \quad J_{\varphi_{jk}} \quad J_{\varphi_{ik}}], \quad k = 1..6 \text{ and } [d\mathbf{L}] = [dL_1 \quad dL_2 \quad \dots \quad dL_6]^T.
\end{aligned}$$

Finally, the dependent errors can be expressed with relation (17):

$$[\delta\Phi] = [\mathbf{J}_\Phi^*]^{-1} [\mathbf{J}_L^*] \cdot [d\mathbf{L}] = [\mathbf{J}^*] \cdot [d\mathbf{L}]. \quad (17)$$

The main problem for expressing analytically the error model consists in reversing the square matrix $[\mathbf{J}_\Phi^*]$ of 30x30 dimension. In the case of Stewart platform, the reverse matrix $[\mathbf{J}_\Phi^*]$ was obtained numerically.

Step 3. Establishment of the and-effector errors

For each open kinematic chain, a set of 5 dependent parameters was used in modelling; in matrix \mathbf{J}^* (relations 17), for each set correspond 5 lines: the first 5 for leg 1, the following 5 for leg 2 a.s.o.

The operational errors of the end-effector can be expressed by replacing the dependent errors expressions (rel. 17) into (rel. 12), with particularization for one of legs. Considering the chain 1, the end-effector errors expressed in the frame \mathcal{R}_p become:

$$\begin{bmatrix} d_x \\ d_y \\ d_z \\ \delta_x \\ \delta_y \\ \delta_z \end{bmatrix}_p = [\mathbf{J}_1] \begin{bmatrix} \delta\gamma_{x1} \\ \delta\gamma_{y1} \\ dL_1 \\ \delta\varphi_{z1} \\ \delta\varphi_{y1} \\ \delta\varphi_{x1} \end{bmatrix} = [\mathbf{J}_1] [\mathbf{J}_{\Phi 1}^*] \begin{bmatrix} dL_1 \\ dL_2 \\ dL_3 \\ dL_4 \\ dL_5 \\ dL_6 \end{bmatrix}, \quad [\mathbf{J}_{\Phi 1}^*] = \begin{bmatrix} J_{\gamma_{x1}}^{L_1} & J_{\gamma_{x1}}^{L_2} & J_{\gamma_{x1}}^{L_3} & J_{\gamma_{x1}}^{L_4} & J_{\gamma_{x1}}^{L_5} & J_{\gamma_{x1}}^{L_6} \\ J_{\gamma_{y1}}^{L_1} & J_{\gamma_{y1}}^{L_2} & J_{\gamma_{y1}}^{L_3} & J_{\gamma_{y1}}^{L_4} & J_{\gamma_{y1}}^{L_5} & J_{\gamma_{y1}}^{L_6} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ J_{\varphi_{z1}}^{L_1} & J_{\varphi_{z1}}^{L_2} & J_{\varphi_{z1}}^{L_3} & J_{\varphi_{z1}}^{L_4} & J_{\varphi_{z1}}^{L_5} & J_{\varphi_{z1}}^{L_6} \\ J_{\varphi_{y1}}^{L_1} & J_{\varphi_{y1}}^{L_2} & J_{\varphi_{y1}}^{L_3} & J_{\varphi_{y1}}^{L_4} & J_{\varphi_{y1}}^{L_5} & J_{\varphi_{y1}}^{L_6} \\ J_{\varphi_{x1}}^{L_1} & J_{\varphi_{x1}}^{L_2} & J_{\varphi_{x1}}^{L_3} & J_{\varphi_{x1}}^{L_4} & J_{\varphi_{x1}}^{L_5} & J_{\varphi_{x1}}^{L_6} \end{bmatrix}, \quad (18)$$

where $J_{t_i}^{L_i}$ is an element of matrix \mathbf{J}^* (relations 17) and represent the influence factor of deviation dL_i on the dependent parameter t_i .

By generalizing the relations 18, the error Jacobean for deviations of active joint variables can be expressed through any of the following relations:

$$[\mathbf{J}_L] = [\mathbf{J}_1] [\mathbf{J}_{\Phi 1}^*] = [\mathbf{J}_2] [\mathbf{J}_{\Phi 2}^*] = \dots = [\mathbf{J}_6] [\mathbf{J}_{\Phi 6}^*]. \quad (19)$$

4.2 Geometrical parameter errors modelling

The influence of deviations of the parameters which define the parallel structure geometry on the end-effector errors is described through a linear model, where the error Jacobean \mathbf{J}_G represents the system matrix:

$$[d\mathbf{X}]_p = [\mathbf{J}_{G_p}] \cdot [d\mathbf{G}], \quad (20)$$

where $[d\mathbf{X}]_p = [d_x \ d_y \ d_z \ \delta_x \ \delta_y \ \delta_z]_p^T$ is the 6 dimensions vector of the operational errors of the end-effector, corresponding to the characteristic point P (Fig. 3), reduced in \mathfrak{R}_p ;

$[d\mathbf{G}] = [\delta\alpha_{z_1} \ \delta\alpha_{y_1} \ dr_{f_1} \ dr_{m_1} \ \delta\beta_{y_1} \ \delta\beta_{z_1} \ \dots \ \delta\alpha_{z_6} \ \delta\alpha_{y_6} \ dr_{f_6} \ dr_{m_6} \ \delta\beta_{y_6} \ \delta\beta_{z_6}]^T$ is the vector of geometric errors.

This modelling is based on the following assumptions:

- Platform's command is considered to be ideal and so the joint errors $d\mathbf{L}$ are null.
- Structure geometry is affected by known and constant in time errors.

In this case, the parallel structure is fictitious splitted into 6 open kinematic chains (OKC), one for each leg: $O_f A_i B_i O_m P$ (see Fig. 3). Further on, the following algorithm, consisting of three main steps, is applied.

Step 1. Description of the end-effector errors for the 6 OKC

In order to model the influence of the geometrical deviations, in the proposed method is used an *equivalent structure*, in which for each geometrical modelling parameter is associated one degree of freedom fictitious joint (prismatic or revolute). Each OKC associated to the Stewart platform becomes a serial structure with 12 one degree of freedom joints (therefore 12 independent parameters – see. Fig. 2): **RRTRRRRRTRR**. Because the end-effector errors will be expressed in the final frame \mathfrak{R}_p , the direct kinematical modelling for finite displacements of OKC will be done with the following homogenous operators of *D-F type (type K)* (Gogu, 1995; Gogu et al., 1997):

$$A_{01} = \mathbf{R}_z(\alpha_{zi}); \ A_{12} = \mathbf{R}_y(\alpha_{yi}); \ A_{23} = \mathbf{T}_x(r_{fi}); \ A_{34} = \mathbf{R}_x(\gamma_{xi}); \ A_{45} = \mathbf{R}_y(\gamma_{yi}) \cdot \mathbf{T}_z(L_i); \ A_{56} = \mathbf{R}_z(\varphi_{zi}); \\ A_{67} = \mathbf{R}_y(\varphi_{yi}); \ A_{78} = \mathbf{R}_x(\varphi_{xi}); \ A_{89} = \mathbf{T}_x(-r_{mi}); \ A_{9_{10}} = \mathbf{R}_y(-\beta_{yi}); \ A_{10_{11}} = \mathbf{R}_z(-\beta_{zi}) \cdot \mathbf{T}_z(l_{mp}).$$

The operational errors for each OKC are described using the general relations for open chains:

$$\begin{aligned} [d_{x_i} \ d_{y_i} \ d_{z_i} \ \delta_{x_i} \ \delta_{y_i} \ \delta_{z_i}]_p^T &= [\mathbf{J}_i] [\delta\alpha_{z_i} \ \delta\alpha_{y_i} \ dr_{f_i} \ \delta\gamma_{x_i} \ \delta\gamma_{y_i} \ \delta\varphi_{z_i} \ \delta\varphi_{y_i} \ \delta\varphi_{x_i} \ dr_{m_i} \ \delta\beta_{y_i} \ \delta\beta_{z_i}]^T \\ &= [\mathbf{J}_i] [d\mathbf{G}_i], \quad [\mathbf{J}_i] = [J_{\alpha_{z_i}} \ J_{\alpha_{y_i}} \ J_{r_{f_i}} \ J_{\gamma_{x_i}} \ J_{\gamma_{y_i}} \ J_{\varphi_{z_i}} \ J_{\varphi_{y_i}} \ J_{\varphi_{x_i}} \ J_{r_{m_i}} \ J_{\beta_{y_i}} \ J_{\beta_{z_i}}]^T, \quad i=1..6, \end{aligned} \quad (21)$$

where the column vectors from matrix \mathbf{J}_i describe the influence of the modelling parameters errors, used in kinematical description of OKC.

Step 2. Identification of dependent errors

Even if there are considered independent in the equivalent structure, the relative displacements from the joints which are not commanded A_i and B_i are dependent displacements and, thus, the displacements errors are also dependent; there expressions can be identified by remodelling the parallel structure through the following condition: *the effector errors are identical for all the 6 OKC*:

$$[d_{x_i} \ d_{y_i} \ d_{z_i} \ \delta_{x_i} \ \delta_{y_i} \ \delta_{z_i}]_p^T = [\mathbf{J}_1] [d\mathbf{G}_1] = \dots = [\mathbf{J}_6] [d\mathbf{G}_6]. \quad (22)$$

Five independent matrix equations are derived from (rel. 22); separating the dependent terms from the independent ones, it results:

$$\begin{aligned} & \begin{bmatrix} J_{\gamma_{x1}} & J_{\gamma_{y1}} & J_{\varphi_{z1}} & J_{\varphi_{y1}} & J_{\varphi_{x1}} \end{bmatrix} \begin{bmatrix} \delta\gamma_{x1} \\ \delta\gamma_{y1} \\ \delta\varphi_{z1} \\ \delta\varphi_{y1} \\ \delta\varphi_{x1} \end{bmatrix} - \begin{bmatrix} J_{\gamma_{zk}} & J_{\gamma_{yk}} & J_{\varphi_{zk}} & J_{\varphi_{yk}} & J_{\varphi_{zk}} \end{bmatrix} \begin{bmatrix} \delta\gamma_{zk} \\ \delta\gamma_{yk} \\ \delta\varphi_{zk} \\ \delta\varphi_{yk} \\ \delta\varphi_{zk} \end{bmatrix} = \\ & = \begin{bmatrix} J_{\alpha_{zk}} & J_{\alpha_{yk}} & J_{r_k} & J_{r_{mk}} & J_{\beta_k} & J_{\beta_{z1}} \end{bmatrix} \begin{bmatrix} \delta\alpha_{zk} \\ \delta\alpha_{yk} \\ dr_{rk} \\ dr_{mk} \\ \delta\beta_{yk} \\ \delta\beta_{zk} \end{bmatrix} - \begin{bmatrix} J_{\alpha_{z1}} & J_{\alpha_{y1}} & J_{r_1} & J_{r_{m1}} & J_{\beta_{y1}} & J_{\beta_{z1}} \end{bmatrix} \begin{bmatrix} \delta\alpha_{z1} \\ \delta\alpha_{y1} \\ dr_{r1} \\ dr_{m1} \\ \delta\beta_{y1} \\ \delta\beta_{z1} \end{bmatrix}, \quad k = 2..6. \quad (23) \end{aligned}$$

The five systems (23) are assembled into one matrix equation:

$$[\mathbf{J}_\Phi^*] \cdot [\delta\Phi] = [\mathbf{J}_G^*] \cdot [d\mathbf{G}], \quad (24)$$

where $[\delta\Phi] = [\delta\gamma_{x1} \ \delta\gamma_{x2} \ \delta\varphi_{z1} \ \delta\varphi_{y1} \ \delta\varphi_{x1} \ \dots \ \delta\varphi_{y6} \ \delta\varphi_{x6}]^T$ is the global vector of dependent errors and

$$[\mathbf{J}_\Phi^*] = \begin{bmatrix} \mathbf{J}_{\Phi1} & -\mathbf{J}_{\Phi2} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{J}_{\Phi1} & \mathbf{0} & -\mathbf{J}_{\Phi3} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{J}_{\Phi1} & \mathbf{0} & \mathbf{0} & -\mathbf{J}_{\Phi4} & \mathbf{0} & \mathbf{0} \\ \mathbf{J}_{\Phi1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{J}_{\Phi5} & \mathbf{0} \\ \mathbf{J}_{\Phi1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{J}_{\Phi6} \end{bmatrix}, \quad [\mathbf{J}_G^*] = \begin{bmatrix} -\mathbf{J}_{G1} & \mathbf{J}_{G2} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{J}_{G1} & \mathbf{0} & \mathbf{J}_{G3} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{J}_{G1} & \mathbf{0} & \mathbf{0} & \mathbf{J}_{G4} & \mathbf{0} & \mathbf{0} \\ -\mathbf{J}_{G1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{J}_{G5} & \mathbf{0} \\ -\mathbf{J}_{G1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{J}_{G6} \end{bmatrix}, \quad (25)$$

$$\mathbf{J}_{\Phi k} = \begin{bmatrix} J_{\gamma_{zk}} & J_{\gamma_{yk}} & J_{\varphi_{zk}} & J_{\varphi_{yk}} & J_{\varphi_{zk}} \end{bmatrix}, \quad \mathbf{J}_{Gk} = \begin{bmatrix} J_{\alpha_{zk}} & J_{\alpha_{yk}} & J_{r_k} & J_{r_{mk}} & J_{\beta_k} & J_{\beta_{z1}} \end{bmatrix}, \quad k = 1..6. \quad (26)$$

Finally, the dependent errors expressions can be expressed by relation:

$$[\delta\Phi] = [\mathbf{J}_\Phi^*]^{-1} [\mathbf{J}_G^*] \cdot [d\mathbf{G}] = [\mathbf{J}^*] \cdot [d\mathbf{G}], \quad [\mathbf{J}^*] = \begin{bmatrix} \mathbf{J}_1^* \\ \vdots \\ \mathbf{J}_6^* \end{bmatrix}. \quad (27)$$

The matrix $[\mathbf{J}_\Phi^*]$ is a square matrix, of 30×30 dimension, and, therefore, reversible.

Step 3. Establishment of effector errors

Matrix \mathbf{J}^* (relations 27) has the dimension 30×36 and it can be split into 6 submatrix \mathbf{J}_i^* of 5 lines, representing the error Jacobean of the dependent deviations from leg i related to deviations $d\mathbf{G}$. Particularized for one of the chains, for instance OKC 1, the end-effector errors expressed in \mathfrak{R}_p become:

$$\begin{aligned} \begin{bmatrix} d_{x_i} & d_{y_i} & d_{z_i} & \delta_{x_i} & \delta_{y_i} & \delta_{z_i} \end{bmatrix}^T &= \mathbf{J}_1 \begin{bmatrix} \delta\alpha_{z1} & \delta\alpha_{y1} & dr_{f1} & \delta\gamma_{x1} & \delta\gamma_{y1} & \delta\varphi_{z1} & \delta\varphi_{y1} & \delta\varphi_{x1} & dr_{m1} & \delta\beta_{y1} & \delta\beta_{z1} \end{bmatrix}^T = \\ &= \mathbf{J}_1 \mathbf{J}_{\Phi_1}^* \mathbf{dG}, \end{aligned} \quad (28)$$

$$\mathbf{J}_{\Phi_1}^* = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 & 0 \\ J_{\gamma_{z1}}^{\alpha_{z1}} & J_{\gamma_{z1}}^{\alpha_{y1}} & J_{\gamma_{z1}}^{r_{f1}} & J_{\gamma_{z1}}^{r_{m1}} & J_{\gamma_{z1}}^{\beta_{y1}} & J_{\gamma_{z1}}^{\beta_{z1}} & \dots & J_{\gamma_{z1}}^{\beta_{y6}} & J_{\gamma_{z1}}^{\beta_{z6}} \\ J_{\gamma_{y1}}^{\alpha_{z1}} & J_{\gamma_{y1}}^{\alpha_{y1}} & J_{\gamma_{y1}}^{r_{f1}} & J_{\gamma_{y1}}^{r_{m1}} & J_{\gamma_{y1}}^{\beta_{y1}} & J_{\gamma_{y1}}^{\beta_{z1}} & \dots & J_{\gamma_{y1}}^{\beta_{y6}} & J_{\gamma_{y1}}^{\beta_{z6}} \\ J_{\varphi_{z1}}^{\alpha_{z1}} & J_{\varphi_{z1}}^{\alpha_{y1}} & J_{\varphi_{z1}}^{r_{f1}} & J_{\varphi_{z1}}^{r_{m1}} & J_{\varphi_{z1}}^{\beta_{y1}} & J_{\varphi_{z1}}^{\beta_{z1}} & \dots & J_{\varphi_{z1}}^{\beta_{y6}} & J_{\varphi_{z1}}^{\beta_{z6}} \\ J_{\varphi_{y1}}^{\alpha_{z1}} & J_{\varphi_{y1}}^{\alpha_{y1}} & J_{\varphi_{y1}}^{r_{f1}} & J_{\varphi_{y1}}^{r_{m1}} & J_{\varphi_{y1}}^{\beta_{y1}} & J_{\varphi_{y1}}^{\beta_{z1}} & \dots & J_{\varphi_{y1}}^{\beta_{y6}} & J_{\varphi_{y1}}^{\beta_{z6}} \\ J_{\varphi_{x1}}^{\alpha_{z1}} & J_{\varphi_{x1}}^{\alpha_{y1}} & J_{\varphi_{x1}}^{r_{f1}} & J_{\varphi_{x1}}^{r_{m1}} & J_{\varphi_{x1}}^{\beta_{y1}} & J_{\varphi_{x1}}^{\beta_{z1}} & \dots & J_{\varphi_{x1}}^{\beta_{y6}} & J_{\varphi_{x1}}^{\beta_{z6}} \\ 0 & 0 & 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & \dots & 0 & 0 \end{bmatrix}, \quad (29)$$

where $J_{t_i}^{p_i}$ represents the influence factor of deviation dp_i on the dependent parameter t_i which is a component part of matrix \mathbf{J}^* (rel. 27).

Generalizing, the error Jacobean for the deviations of joint variables can be expressed through any of the following relations:

$$\mathbf{J}_G = \mathbf{J}_1 \mathbf{J}_{\Phi_1}^* = \mathbf{J}_2 \mathbf{J}_{\Phi_2}^* = \dots = \mathbf{J}_6 \mathbf{J}_{\Phi_6}^*. \quad (30)$$

4.3 The error model for geometrical and kinematical parameters deviations

The complete kinematical error model, which includes both the influences of the active joint variables L_i and of the geometrical parameters can be deduced through a similar approach; the only changing in the modelling prerequisites is referring to the fact that the displacements deviations from the commanded joints have to be included between the error sources of Stewart platform. Therefore, the number of error source parameters is changing from 36 to 42.

In the first step of the modelling algorithm is included also deviation dL_i generated when the relative displacements from the actuated joints C_i are commanded. In step 2 the 30 dependent deviations are identified in relations with the 42 independent variables. In this case is also necessary to reverse the matrix of 30×30 dimension.

Finally, the end-effector errors expressed in reference frame \mathcal{R}_p can be deduced by replacing the expressions of the dependent errors in the error model of one of the OKC.

In conclusion, the complete error model can be obtained by assembling the previous partial models:

$$\begin{bmatrix} d_{x_i} & d_{y_i} & d_{z_i} & \delta_{x_i} & \delta_{y_i} & \delta_{z_i} \end{bmatrix}^T = \mathbf{J}_{1G} \mathbf{J}_{\Phi_{1G}}^* \mathbf{dG} + \mathbf{J}_{1L} \mathbf{J}_{\Phi_{1L}}^* \mathbf{dL} = \mathbf{J}_G \mathbf{dG} + \mathbf{J}_L \mathbf{dL}. \quad (31)$$

5. Accuracy numerical simulations

The numerical simulation of the error model for Stewart-DELTALAB platform had the following objectives:

1. *Validation of the error model* by verifying the results obtained in the numerical and graphical simulation. Thus, for a set of representative configurations of Stewart platform were applied the known deviations of source-parameters and the end-effector errors were calculated (rel. 18). The configurations tested with error, were generated with a graphical tool (in this case AutoCAD) and were established the effective values of the modeling independent and dependent parameters, using specific functions. In all the tested variants, the simulation showed the correctness of the elaborated models.
2. *Identification of the end-effector errors* on a given trajectory, for specified values of the source errors. Using numerical simulation, the global effect of error sources and the importance of each error parameter on the positioning and orienting precision of end-effector can be calculated. In this way, can be identified the factors with a maximum influence and thus recommendations for constructive and functional design can be elaborated.

The results of numerical simulation of the precision model for the case of a linear trajectory, given through the start configuration $(x_m, y_m, z_m, \theta_1, \theta_2, \theta_3) = (-100\text{mm}, 100\text{mm}, 100\text{mm}, 0^\circ, 0^\circ, 0^\circ)$ and -final $(100\text{mm}, -100\text{mm}, 200\text{mm}, 45^\circ, 30^\circ, -30^\circ)$, are presented in Figure 4. The represented end-effector errors were generated considering that all active joint errors $\Delta L_i = 1\text{ mm}$. Thus, the positioning on axis z is achieved with the biggest deviations (Fig. 4,a), while the maximum angular deviations are registered on axis y (Fig. 4,b).

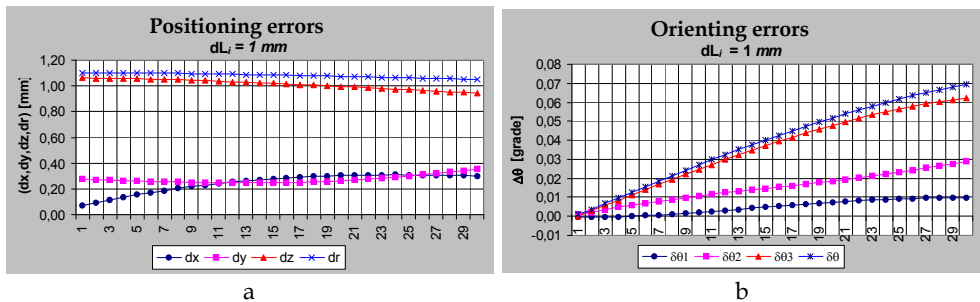


Fig. 4. Numerical simulation of the joint error model: end-effector positioning errors (a) and orienting errors (b) obtained for joint errors $\Delta L_i = 1\text{ mm}$

The objective of the numerical simulation of the geometric error model was to identify the end-effector errors on a given trajectory, for the specified values of source errors (Fig. 5). Thus, it can be established both the global effect of all the error sources and the importance of each error parameter on the positioning and orienting precision of the end-effector; therefore, the factors with a maximum influence can be identified and the recommendations for constructive and functional design can be elaborated.

Considering the former trajectory, the end-effector errors were generated in the assumption that all the linear geometrical parameters have 1 mm deviations, while the angular ones

have 1° deviation. The following conclusions and recommendations can be formulated analyzing the graphical representations of the end-effector errors (Fig. 5):

- Referring to the positioning precision it can be noticed the superior influence of the parameters α_{yi} (Fig. 5,a) and β_{yi} (Fig. 5,b) similar to the relatively reduced effects of parameters r_{fi} (Fig. 5,c) and r_{mi} (Fig. 5,d).
- The orienting precision depends in a small measure of the deviations of parameters r_{fi} (Fig. 5,e) and r_{mi} (Fig. 5,f). On the other hand, the orienting precision is more dependent on the deviations of parameters α_{zi} and β_{zi} .

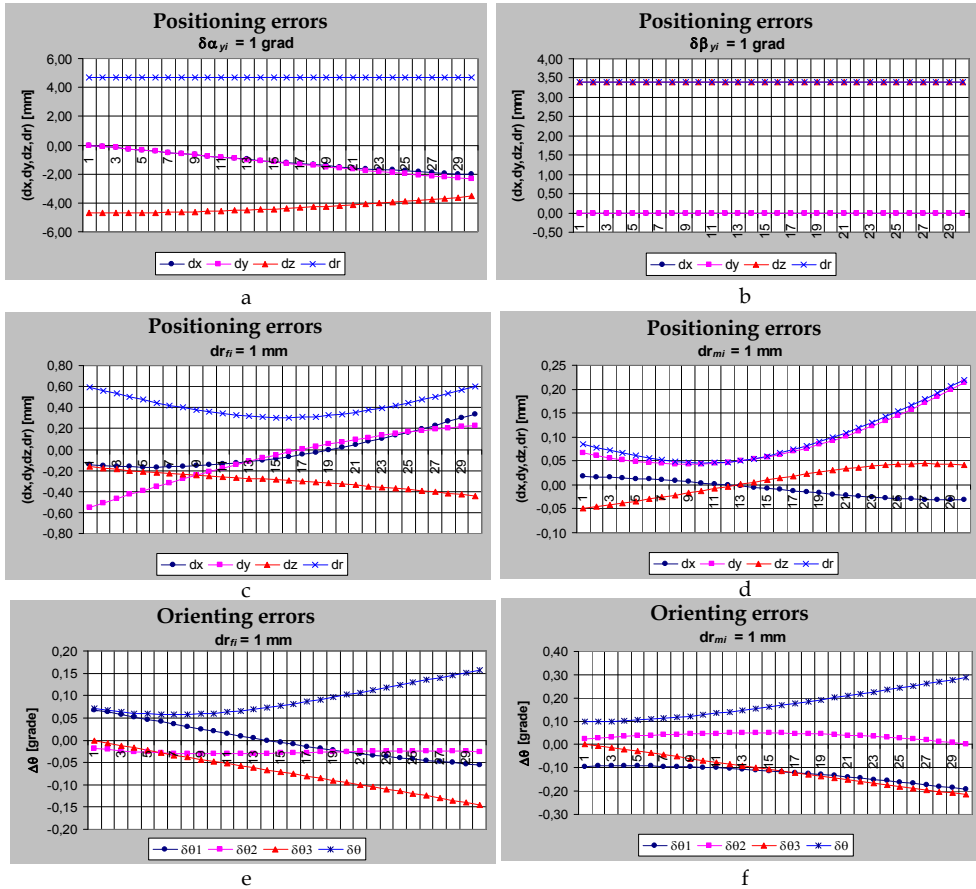


Fig. 5. The influence of some geometric parameter errors on the trajectory end-effector errors

- From the numerical study of the precision of Stewart platform on the mentioned trajectory, it can be formulated the recommendation that the maximum precision for the angular parameters (α_{yir} , β_{yir} , α_{zir} , β_{zir}) has to be assured.

6. Accuracy experimental validation

Concerning the Stewart DELTALAB platform error modelling, some explanatory notes and the necessary notations in the experimental testing are presented further on:

- two categories of reference frames are associated to the tested platform: *theoretical reference frames* (used in the platform command program) – associated to the theoretical plane given by centres of the spherical joints of mobile platform (points B_i), and *measure frames* (used in the measure process) – associated to a real surface of the mobile platform;
- the taken measurements were of *relative type*, in relation to a reference frame defined by the 3D measurement machine, on the base of the measure frame for the reference position of the platform;

- $\mathfrak{R}_{mexp}(O_{mexp}x_{mexp}y_{mexp}z_{mexp})$ – the *theoretic* frame associated to the mobile platform in an experimentally specified position;
- $\mathfrak{R}_{pref}(O_{pref}x_{pref}y_{pref}z_{pref})$ – the *measurement* frame associated to the mobile platform in a reference position;
- $\mathfrak{R}_{pexp}(O_{pexp}x_{pexp}y_{pexp}z_{pexp})$ – the *measurement* frame associated to the mobile platform in an experimentally established position.

In order to become possible and to offer complete data on the accuracy, a first step in experimental testing is to identify the constructive elements of the platform and the accessories. In this context, the following explanatory notes are made:

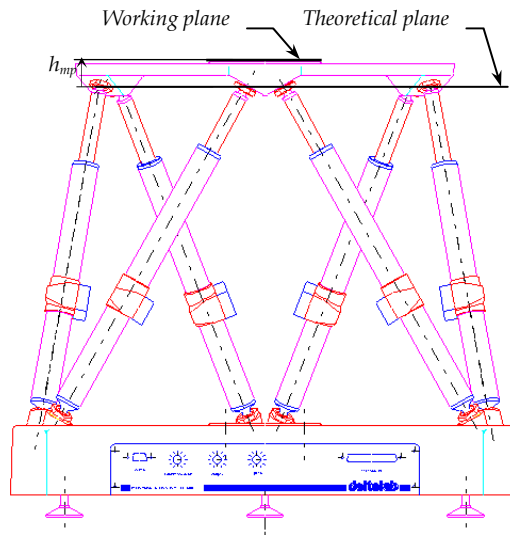


Fig. 6. Theoretical and working planes of Stewart platform

1°. Plane $x_m O_m y_m$ of frame \mathfrak{R}_m is identical to points B_i plane; being a fictive plane, in experimental research was used frame \mathfrak{R}_p for which plane $x_p O_p y_p$ is materialized by a plane finished surface (*working plane* – see Fig. 6). The frame \mathfrak{R}_p is parallel to \mathfrak{R}_m and is obtained through a translation with the distance $h_{mp} = O_m O_p = 40.55$ mm on the $O_m z_m$ axis.

2°. The Stewart platform includes, as additional accessories, two cylindrical finished bolts, assembled on the working plate in point O_p and, respectively, in a point on the axis $O_p x_p$ (Fig. 3 and 4). In this way, frame \mathfrak{R}_p is materialized as follows:

- axis $O_p z_p$ through the normal to the working plane;
- the origin O_p as the intersection point of working plane and the axis of the bolt which is assembled in this point;
- axis $O_p x_p$ as the line described by 2 points: O_p and the intersection point of the second bolt axis and the working plane;
- axis $O_p y_p$ with, implicitly, the unit vector $\vec{j}_p = \vec{k}_p \times \vec{i}_p$.

6.1 The 3D measurement machine TEMPO

The *Tri-Measures* machine (Fig. 7) is metrology equipment with high performances. From the structural point of view, the machine is assimilated to an orthogonal robot of portal type, with three independent axes. The final element, which performs a translation on vertical (z axle), has an orientation measurement head at its extremity, that has a sensing head system (Fig. 7).

The machine is characterized by a high rigidity, its mobile parts moving on an aerostatic cushion and a high geometrical precision.

The logistic administration of measurement machine functioning is obtained through the program METROSOFT 3D. The program allows to measure parts with plane, spherical, cylindrical or conical surfaces. The program is handling almost exclusively through a control desk, with specialized keys for different categories and command types.

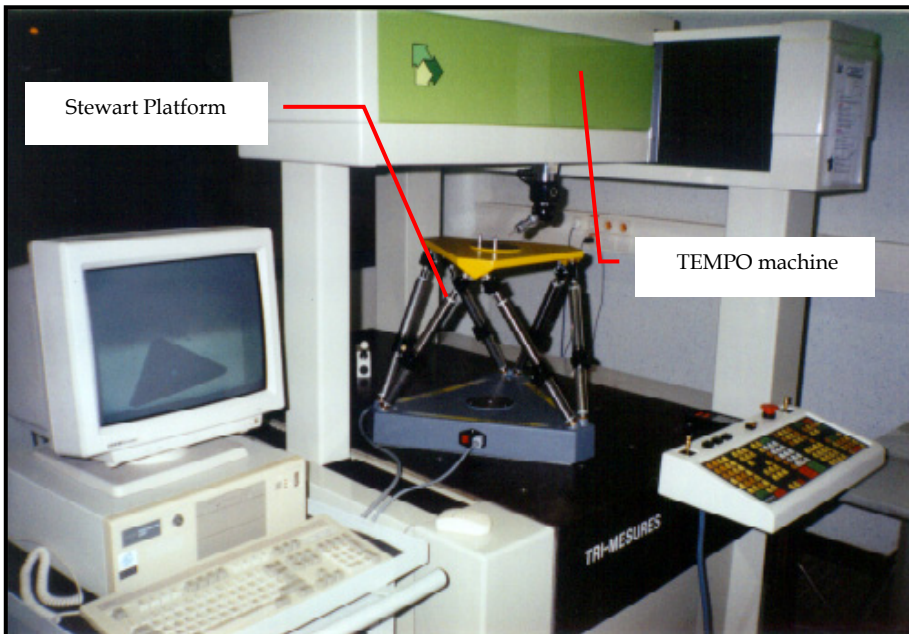


Fig. 7. The 3D Measurement TEMPO machine and the Stewart-DELTALAB platform

6.2 The experimental research program

The experimental research had in view to fulfil a program based on the following objectives:

1. Establishment of Stewart platform *repeatability*.
2. Establishment of absolute precision (*accuracy*).
3. The experimental testing for proposed precision models validation.

The experimental testing was preceded by identification of a minimum number of parameters (points and vectors) which have to be measured and which allow the analytical establishment of real position of mobile platform (of the reference frame \mathfrak{R}_p).

Starting from the experimental and command values, the actual operational errors of the Stewart platform are established, with their help being quantified the accuracy and repeatability.

Therefore, the main steps in the measurement process of Stewart platform accuracy are presented:

1. The Stewart platform is placed on the measurement machine working table and runs the platform command program.
2. The platform is put under tension and is commanded to move to the initial position (*zero position*).
3. After starting up the measurement machine, the necessary configuration of the measurement head is calibrated through filling a standard sphere and their memorization.
4. A *frame part* is defined, materializing the reference frame \mathfrak{R}_p . This is achieved in 3 phases:
 - a. Establishment of a *primary direction*, which gives one of the frame part axes.
 - b. Establishment of a *secondary direction*, which materializes the second axle of the part frame.
 - c. Specification of the origin of the frame part.
5. The platform is moved in the testing pose.
6. Are taken measurements in order to establish the frame \mathfrak{R}_{exp} position (Fig. 8):
 - a. *Measurement of working plane*. The procedure imposes a *plane* command selection and filing of 4 points (4 is the implicit value, which can be modified and represents the minimum accepted number of points). There are supplied to the user the components of the unit vector normal to the plane.
 - b. Fulfil of the two cylinders, placed in the two points P_1 and P_2 , through *cylinder* command initialization and fulfil of minimum 9 points per cylinder. These cylinders were materialized through calibrated bolts with $\varnothing 8 \times 25$ dimensions.
 - c. The two points P_1 and P_2 are obtained as intersections of the two cylinders with working plane. The coordinates of the two points are displayed in the part system declared active.
 - d. Identification of O_x axle as a line which passes through points P_1 and P_2 . It is used the *connex* command and it is given the axle by the unit vector.

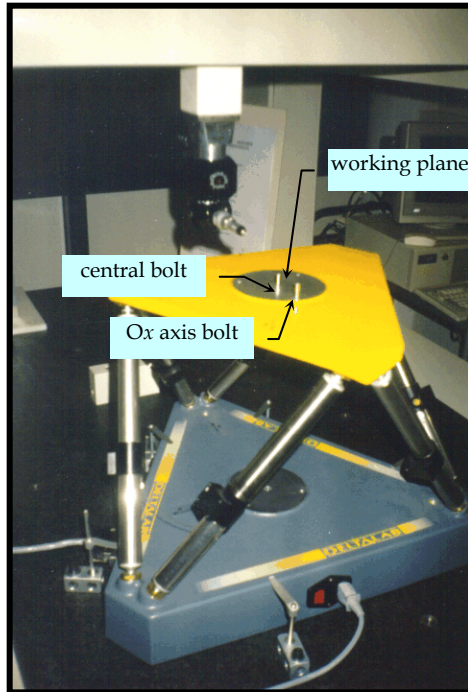


Fig. 8. Geometric elements used in the experimental research

Explanatory notes:

- The machine can report measured values and can make calculus both in the machine frame and in the frame defined by the user. The machine reference frame is defined with the axis parallel to the displacement directions of mobile elements (x axle - the longitudinal axle, y - transversal and z - vertical, see Fig. 7) and the origin in the standard sphere centre. A user frame is defined in accordance to point 4 and is associated to the geometrical form of a part; this will be named further on as *frame part*.
- At the intersection of a plane with a cylinder, the machine program obtains a point and not an ellipse (circle), considering the intersection between the plane and the cylinder axle.

The algorithm for actual operational errors calculus, corresponding to the relative measurements, is based on scheme from Figure 9.

The final purpose of the mathematical processing of experimental values is to identify the 6 dimensions vector of the actual operational errors as a measure of the difference between the real position (experimentally established) and the commanded one (theoretical). So, first are established the expression of the homogenous operator $\mathbf{A}_{p \rightarrow exp}$ for each measurement; in phase 2 are identified the real errors $\mathbf{A}_{p \rightarrow exp}$, related to frame \mathfrak{R}_p .

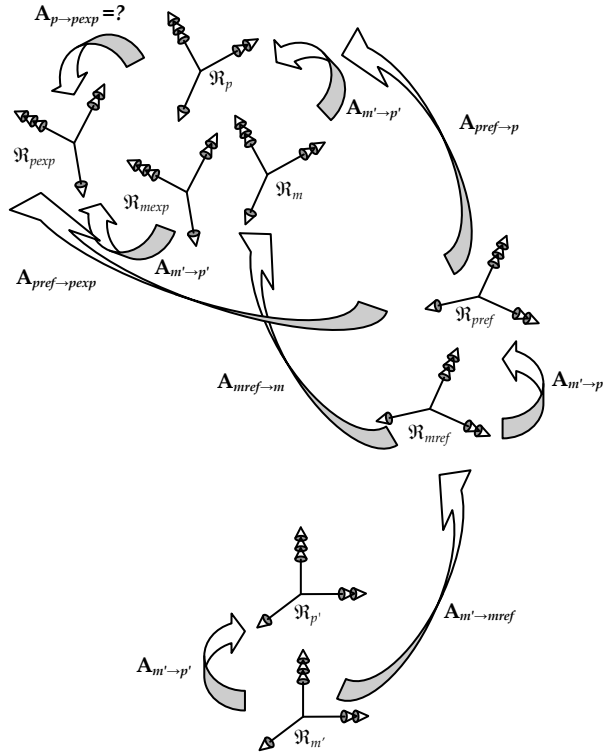


Fig. 9. Reference frames associates to the Stewart platform

The selection of frame \mathfrak{R}_p as a reference frame (and not of \mathfrak{R}_m) is justified through the physical existence of working plane (as a measurement plane); this interferes as a location element in the Stewart platform tasks, unlike frame \mathfrak{R}_m , which is fictive.

In the relative measurements case, the measured values are reported to a part frame defined in a reference position \mathfrak{R}_{pref} .

– There are known (Fig. 9):

○ $\mathbf{A}_{m' \rightarrow p'}$, $\mathbf{A}_{m' \rightarrow m}$ și $\mathbf{A}_{m' \rightarrow mref}$;

○ $\mathbf{A}_{p' \rightarrow pref} = (\mathbf{A}_{m' \rightarrow p'})^{-1} \cdot \mathbf{A}_{m' \rightarrow mref} \cdot \mathbf{A}_{m' \rightarrow p'}$;

○ $\mathbf{A}_{pref \rightarrow pexp}$, established by means of experimental values, related to \mathfrak{R}_{pref} ;

– There is identified the homogenous operator $\mathbf{A}_{p \rightarrow pexp}$ from the following equality (see Fig. 9):

$$\mathbf{A}_{pref \rightarrow pexp} = \mathbf{A}_{pref \rightarrow p} \cdot \mathbf{A}_{p \rightarrow pexp} \Leftrightarrow \mathbf{A}_{p \rightarrow pexp} = (\mathbf{A}_{pref \rightarrow p})^{-1} \cdot \mathbf{A}_{pref \rightarrow pexp}$$

$$\text{where } \mathbf{A}_{m' \rightarrow p'} \cdot \mathbf{A}_{pref \rightarrow p} = \mathbf{A}_{mref \rightarrow m} \cdot \mathbf{A}_{m' \rightarrow p'} \Rightarrow \mathbf{A}_{pref \rightarrow p} = (\mathbf{A}_{m' \rightarrow p'})^{-1} \cdot \mathbf{A}_{mref \rightarrow m} \cdot \mathbf{A}_{m' \rightarrow p'}$$

$$\mathbf{A}_{mref \rightarrow m} = (\mathbf{A}_{m' \rightarrow mref})^{-1} \cdot \mathbf{A}_{m' \rightarrow m}$$

6.3 The experimental validation of the error models

The accuracy model represents the mathematical expression of the dependencies between the operational errors and the source-errors. From the tests on Stewart platform it was pursued the accuracy model validation, considering as source deviations the relative displacements from the actuated joints.

Validation of the precision model consists of:

- The platform is moved in the tested reference positions;
- A part frame is defined in \mathfrak{R}_{pref} (the relative measurements case);
- The platform is moved in adjacent positions, by commanding displacements L_i bordered to those corresponding to \mathfrak{R}_{pref} ;
- The real position of the platform is measured and identified;
- The theoretical and real deviations are established (with the precision model).

For the experimental validation of precision model were selected several representative test-configurations. For one of them, the expression of the error Jacobean, in accordance to (rel. 19), is:

$$J := \begin{bmatrix} -.08237 & .05989 & -.5176 & .6115 & .6305 & -.5668 \\ .7215 & -.7201 & -.3781 & .4234 & -.2856 & .2719 \\ .1949 & .1699 & .2631 & .1183 & .1485 & .2468 \\ -.0002606 & .0002159 & .002222 & .001055 & -.001426 & -.001923 \\ -.001962 & -.001920 & .0007583 & .001220 & .001377 & .0006450 \\ .001861 & -.001910 & .001614 & -.001863 & .001675 & -.001583 \end{bmatrix}.$$

As it can be seen in Figure 10, there is a good concordance between the values given by the precision model and the finite displacements, established on the theoretical model (by numerical simulation) and the experimental values.

The deviations from Figure 10 have the following meanings:

- the "exact" deviations were established with the direct kinematical model for finite displacements, in which were included the final values (corrected) of the kinematical variables $L_i = L_{iref} + \Delta L_i$. The "exact" deviations are theoretical deviations, defined by finite displacements from the reference-test configuration \mathfrak{R}_{pref} to the commanded configuration with errors \mathfrak{R}_p ;
- the *experimental deviations* express the difference between the measured configuration \mathfrak{R}_{exp} and the reference one \mathfrak{R}_{pref} ;
- the *calculated deviations* are obtained applying the linear error model, in which the error Jacobean J corresponds to the test-configuration \mathfrak{R}_{pref} .

The differences between the results obtained through the proposed model and the "exact" model are explained by the linear nature of the precision model (the infinitesimal displacements are level 1 approximations of the finite displacements); these differences tend to zero only for the small values (infinitesimal) of input parameters (source errors) and, respectively, increase for values from the finite domain of inputs in the model.

7. Conclusion

- The proposed modelling method allows deriving the error model through a systemic and algorithmic approach and it is applied for parallel structures of any complexity.
- The analytical error model of Stewart-DELTALAB platform has a relatively high complexity, due to the fact that a matrix of 30×30 dimension has to be reversed; the problem was solved numerically.

- The Jacobean J_L can be numerically expressed. Numerical simulation was used for checking the correctness of the algorithm and of modeling. The error model was also verified through graphical simulation, using AutoCAD.
- The relations for the Jacobean J_G were used, through numerical simulation, to verify the correctness of the algorithm and of the modelling.
- The results, offered by the precision models are in a good concordance both with the values given by numerical simulation and, also, with the experimental values (Fig. 10).

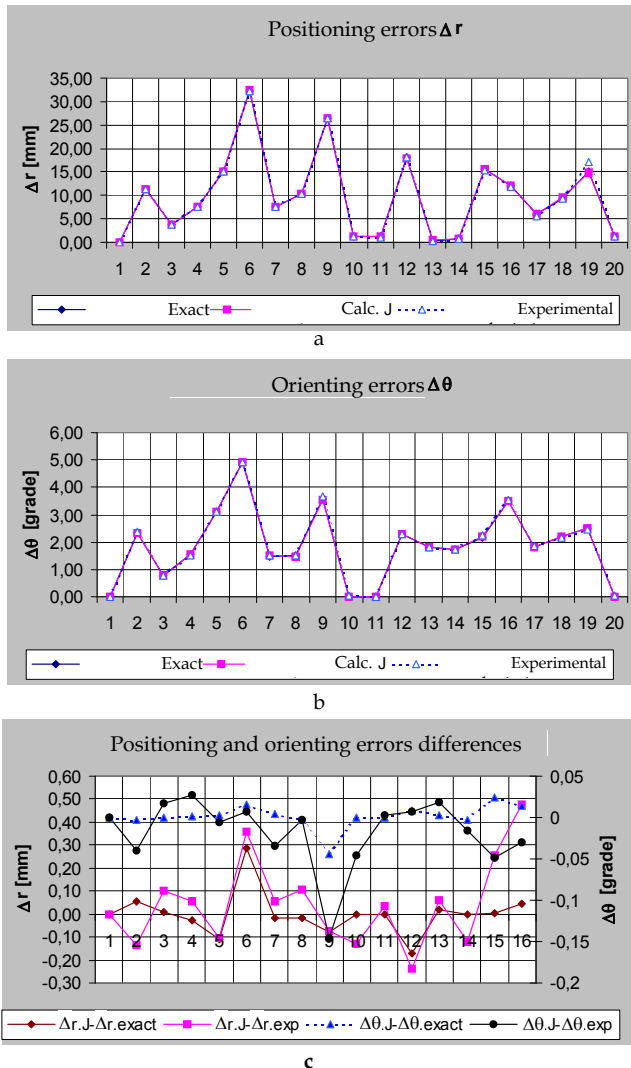


Fig. 10. Experimental validation of theoretical models

- The differences between the results obtained with the proposed model and the experimental values are relatively small (of 10^{-1} mm, respectively of 10^{-1} degrees) and are enclosed in the error limit given by the accuracy error of tested Stewart platform.
- The experimental testing validates the precision models; the conclusion is that the proposed models are correct and the algorithm and the numerical implementation of the models are, also, correct.

Acknowledgment

The authors would like to acknowledge the support of French Institute of Advanced Mechanics (IFMA), Clermont-Ferrand, France in assuring full access to the robotic facilities for experimental researches in this work. The work was partially supported by Romanian project SMART 72197/1.10.2008.

8. References

- Benea, R. (1996). *Contribution à l'étude des robots pleinement parallèles de type 6R-RR-S*. PhD thesis. Université de Savoie, France.
- Briot, S. & Bonev, I. (2008). Accuracy analysis of 3-DOF planar parallel robots, *Mechanism and Machine Theory*, Vol. 43, pp. 445-458.
- Cui, H.; Zhu, Z.; Gan, Z. & Brogardh, T. (2008). Error modeling and accuracy of TAU robot, In: *Parallel Manipulators. New Developments*, Ryu, J.H. (ed.), pp. 269-286, Vienna: I-Tech Education and Publishing, ISBN 978-3-902613-20-2.
- Fichter, E.F. (1986). A Stewart Platform-Based manipulator: General Theory and practical construction. *The International Journal of Robotics Research*, Vol. 5, no.2, pp. 157-185.
- Gogu, Gr. (1995). *Optimisation of the industrial robot kinematics modelling (in Romanian)*. PhD thesis. Transilvania University of Braşov.
- Gogu, Gr.; Coiffet, Ph. & Barraco, A. (1997). *Représentation des déplacements des robots*, Ed. Hermes, Paris.
- Lee, J.; Duffy, J. & Keler, M. (1999). The optimal quality index for the stability of in-parallel planar platform devices. *Journal of Mechanical Design*, Vol.121, pp. 15-20.
- Merlet, J.P. (1990). *Les robot parallèles*, Ed. Hermes, Paris.
- Merlet, J.P. & Gosselin, C.M. (1991). Nouvelle architecture pour un manipulateur parallèle à six degrés de liberté. *Mechanisms and Machines Theory*, Vol. 26, No. 1, pp. 77-90.
- Neagoe, M. (2001). *Contributions to the study of industrial robot precision (in Romanian)*. PhD thesis. Transilvania University of Braşov.
- Paul, R. (1981) *Robot manipulators: mathematics, programming and control*. The MIT Press.
- Pott, A. & Hiller, M. (2008). Kinematic Modeling, Linearization and First-Order Error Analysis, In: *Parallel Manipulators. Towards New Applications*, Wu, H. (ed.), pp. 155-174, I-Tech Education and Publishing, ISBN 978-3-902613-40-0.
- Toyama, S. & Hatae, S. (1989). Error analysis of platform type of robot by means of screw algebra, *Proceedings of the 20th ISIR Int. Symp. on Industrial Robots*, pp. 635-642, Tokyo, Japan.
- Wu, J. & Yin, Z. (2008). A Novel 4-DOF Parallel Manipulator H4, In: *Parallel Manipulators. Towards New Applications*, Wu, H. (ed.), pp. 405-448, Vienna: I-Tech Education and Publishing, ISBN 978-3-902613-40-0.

Effective knowledge acquisition by means of teaching strategies

Marek Woda
Wroclaw University of Technology
Poland

1. Introduction

Despite of the fact that e-learning already proved its great usefulness, it still suffers from many childlike deficiencies. We can point among the other things, the lack of the coherent vision for learning process accomplishment, practical guidelines how to organize consistent learning content (Woda & Walkowiak, 2004). Due to these disadvantages e-learning is being perceived ambiguously and usually incorrectly implemented in real life e-systems that finally lead to limitation of its reliability. Usually, in real world, theory and practice are not on the par, exactly same situation can be observed in the theory of e-learning and its implementations. In the e-learning, whole stress was put on the learning theory, and there are no restrictions or even practical guidelines present in field of technology used for implementations, best implementation practices, which in many cases has negative influence on newly developed e-systems (Woda & Walkowiak, 2008). Currently most of the academics, and schoolteachers noticed the need of standardization and rationalization of this type of teaching.

Not infrequently, knowledge acquisition process in e-learning has a way worse effectiveness than traditional one that takes place in a conventional teaching – and this is especially noticeable in case of the students that are not very proficient in computers. Main cause of this phenomenon is inability to select essential information by students from so-called “informational noise” and the lack of the direct contact with a tutor and/or learning materials have been prepared in inappropriate way by the course organizers.

Focusing only on a knowledge delivery problem in e-learning systems, we can find course material selection with relation to expertise level of a particular student as a main shortcoming. The other, also major drawback is an immense burden for the course administrators, when number of course students exceeds a few dozen or so. Then a number of people who are involved in planning, control, scheduling of classes and students’ progress assessment, increases in proportion to a number of students. Effectiveness of knowledge acquisition is a function of different forms, methods and variety of teaching methods (Nichols, 2008). Nowadays, in a computerization era, teaching effectiveness in e-systems may increase, only when appropriate steps are undertaken along with an application of classical forms and methods teaching, leading to a construction of suitable teaching structures, which are integrated with latest technologies combined with the formal ways of presentation (Woda, 2008).

Teaching technology is an interdisciplinary discipline about education efficiency, pursuing the answer for the question, how to educate quicker, faster, better and less expensive in a defined conditions.

Interdisciplinary nature of the discipline relies on that, it draws its subject of the interest and research methods other disciplines like computer science, cybernetics, theory of systems and communication theories.

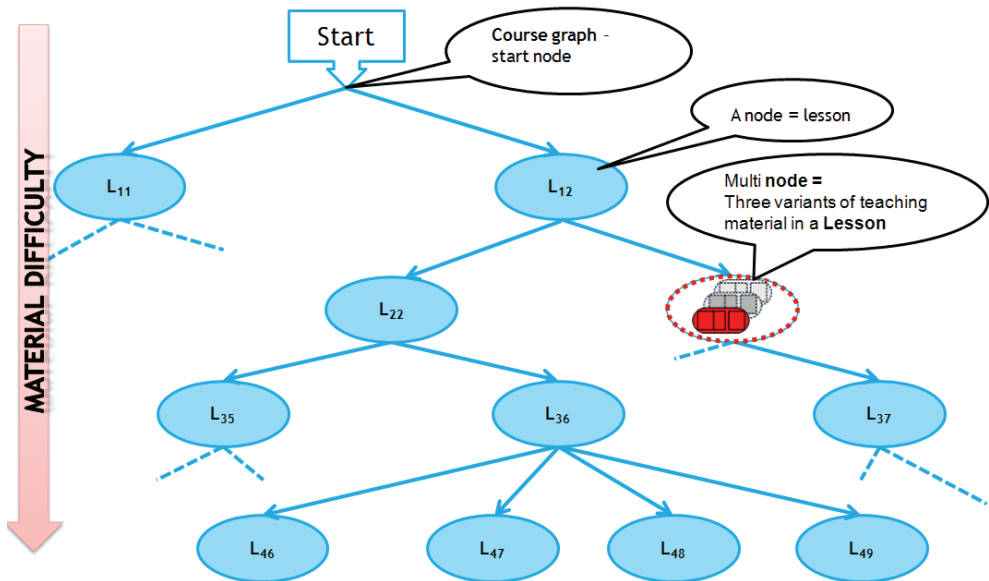


Fig. 1. Teaching material represented as a learning tree

In traditional method learning, human factor is responsible for entire teaching process (Woda & Walkowiak, 2004). This may lead to situations (and usually does), like loss of control over learning progress (due to e.g. mental fatigue of a teacher, badly adapted teaching materials to the students' skills), which finally results in loss of student's attention and willingness to learn (Mghawish & Woda & Michalec, 2006). Nonetheless, excluding completely "human factor" is not possible, and at the same time from a teaching perspective very disadvantageous factor (Al-Dahoud & Walkowiak & Woda, 2008), and it is tightly connected with students feeling of being alienated and which lead to a loss of the control.

The remedy for the presented above distant learning inconveniences and a way to improve efficiency of knowledge acquire process could be application of intelligent system that is driven by smart teaching algorithms (Baloian & Motelet & Pino, 2003, Capusano & Marsella & Salerno, 2000, Dinosereanu & Salomie, 2003, Mghawish & Woda & Michalec, 2006).

2. Teaching strategies

Teaching strategies are algorithms that support navigation within a learning path, during knowledge acquisition process by a student. These algorithms are responsible for directing students on the suitable lesson's variants in the nodes of the learning path (Woda, 2006). Appropriate assignment made by the strategies is being made in a way that teaching material is being selected to suit more adequately student's expertise level, and what is more his ability to learn, according to the criteria.

Navigation algorithms have to lead a student or a group of students thru learning path from first node (starting phase), to another, until end of learning path is reached. State after starting phase is named adaptive state (phase) and it lasts to the end of knowledge acquisition process.

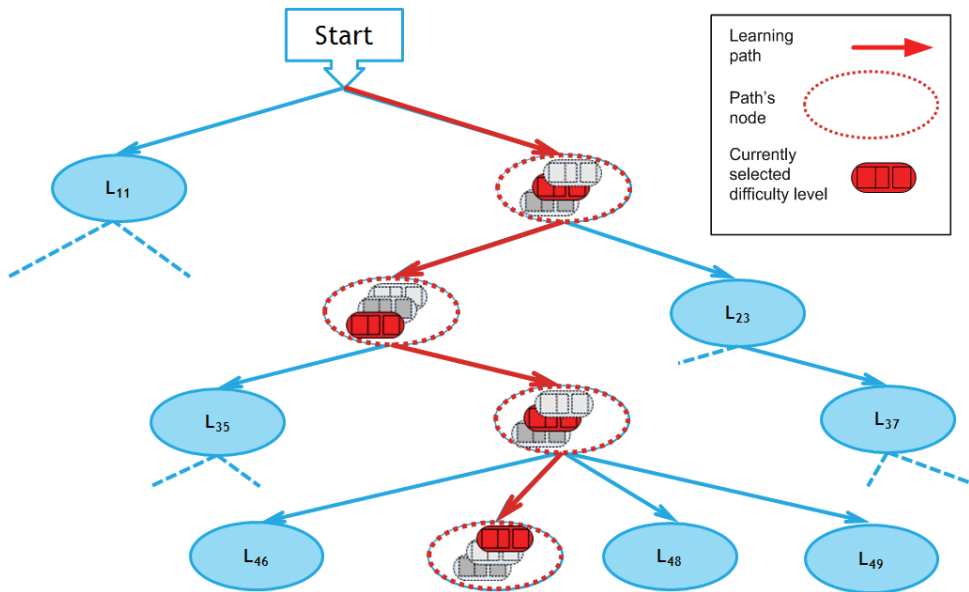


Fig. 2. A learning path with selected nodes (lesson's variants) in the learning tree

Teaching algorithm operation is based on the learning adaptation mechanisms, where knowledge absorption process is being scrutinized on the fly, and historical data, about lessons learnt and scores achieved, are being taken into account in order to assign precise and adequate learning material in a next node from the learning path to achieve best possible (optimal) knowledge acquisition. Learning adaptation means drawing conclusions out of gathered historical data, and then based on them learning "parameters" tuning to match optimal learning pace and form, for a student.

Main task for an adaptive teaching algorithm is to assign each student from a group, appropriate lesson difficult variant (in current the lesson), in order to achieve best result (a note) in a competence test after the lesson. Best result, means required /set by the teacher at the beginning of learning process, usually it is a combination of notes (after a lesson) and credit points (received upon completion of the most difficult lesson variant in a node).

Additionally, at the beginning of learning process one should pay special attention to verify student's initial expertise level, to assign base lesson variant in a start node (starting phase) to match student capability to learn. If the initial expertise is not detected well, it will affect learning efficiency later on, during learning progress (in the adaptive phase). During the adaptive phase, one of available learning strategies, is being assigned to a student, based on his initial expertise level so when an inaccurate strategy is chosen, it greatly affects learning progress and its effectiveness. Lesson difficulty factor is correlated with student's ability to comprehend given material.

3. Learning progress verification

The verification of the knowledge acquisition during duration of a course, takes place after each lesson, in a competence test. It is essential, in order to go to next lesson to firstly pass based current lesson variant (usually least complicated one) in a node. Each student is assessed in a competence test and note is being assigned afterwards and it defines knowledge absorption factor for a lesson. Upon a lesson completion on a specific difficulty factor (lesson variant) student receives a number of credit points, which reflects how the variant was elaborated. Note's value that qualifies student to pass to next lesson is strictly dependent on his base expertise level and lesson's variant, which defines also current competence test.

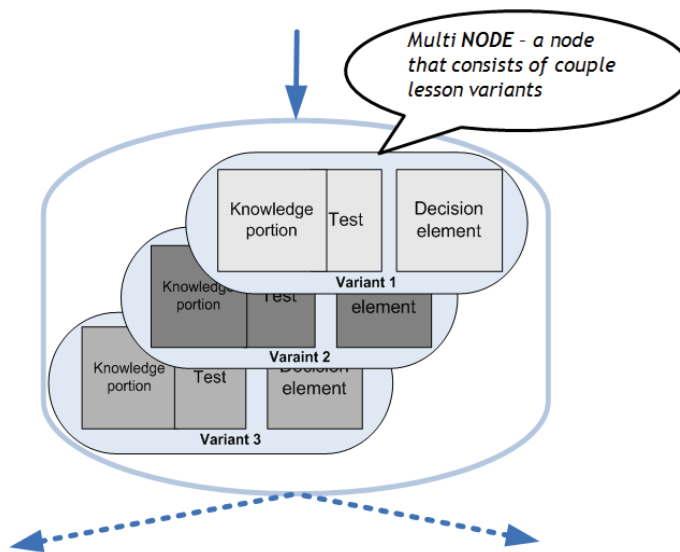


Fig. 3. A multi node of the learning path - variants of the lesson.

The validation of the knowledge acquisition is being done for both a student and for a group (all course participants), after each lesson passed in a learning path (within competence test). Validation procedure has to substantiate (according to taken assumptions), that in fact, strategies of the learning adaption influence on improved knowledge acquisition during learning phase.

The notes received in a competence test, serve as an input data for the strategies. Historical data are taken from student's records (notes, received in previously passed competence tests along with a sum of credit points), which constitute a base for the final assessment of learning progress quality.

In order to be eligible to pass to a next lesson, student must, at least, pass thru easiest material variant in a current lesson node (current competence test must be completed with an acceptable note value) and receiving, at least, one credit point.

4. Learning effectiveness

One of the most common e-learning problems is lower than expected effectiveness of the knowledge absorption. Author, in this chapter focused mainly on that issue. Author concentrated on increasing learning effectiveness by means of teaching strategies. It comes down to best teaching algorithm assignment either for a student or a group from a set of possible adaptive learning strategies. Each strategy ought to meet to given down below criteria.

These strategies ought to organize learning process in a way that best suits student's abilities to learn and finally improve the results (grades and credits) received.

Main criterion, taken into consideration during assessment of learning effectiveness improvement process, is striving after receiving best possible grades and at the same time, more credits. These are two opposing criteria. Receiving best grades could be achieved in an easy way by assignment in every node easiest lesson variant, however it would result in receiving less credits than expected afterwards. Receiving greatest number of credits is only possible once the most elaborated lesson variants are being assigned and finished in the node. Aforementioned facts allow us to clearly assess the quality of teaching strategies.

Other characteristics that prove usefulness of a particular strategy is a total number of students (from a test group) that received at least a half possible credit points during entire learning process. Grades average in a learning process is a supporting factor.

Strategies also strive after fulfilling given below assumptions:

- minimize number of students that cannot comply the optimal learning postulate (minimize drop out from learning process – do not complete competence tests on a required level)
- detect students with incorrectly detected initial expertise level (especially in a start phase)
- “exploit” best students – to assign them more difficult lesson's variants

As a criteria for starting phase strategies quality assessment following have been taken into account:

- a percentage of students that have received more than half credits possible, number of grades scored above group's average, group grade's average, lowest and highest student grade's average

As a criteria for adaptive phase strategies quality assessment following have been taken into account:

- an expertise level distribution changes in time (compare before and after learning), sum of credits after learning process is over, student/group grade's average, number of students that finished learning with more than half possible credits to earn

Each strategy, should meet at least one of aforementioned (main) criteria. Besides each strategy have its own characteristics e.g. global optimum strategy strives to find strategies should be able to find students that are more talented (once found – they are being assigned more difficult / challenging task / lessons). Each strategy acts in a different way based on a student profile detected during learning process. For example minimalistic strategy, strives after assignment students easiest possible lesson's variant in a learning path.

6. Results

All the results have been received in a simulated environment. There were three groups of students, each 100 student, tested. Each group had different base student's expertise level (refer to the Table 1.) and it was served with the same amount of the lesson nodes (50). Simulation was divided in two phases: start phase and adaptive one, where groups were governed by the learning strategies.

Strategy	[%] of students in tested group with better credit points average than average of credit points for a group		
Conservative	19%	22%	27%
Optimal	41%	42%	48%
Simplistic	17%	12%	21%
Reference	37%	44%	49%
Tested group			
	Test Group (1)	Test Group (2)	Test Group (3)
Strategy	The average grades of tested group after the learning process is over		
Conservative	0,4064	0,5054	0,6441
Optimal	0,4528	0,5308	0,6361
Simplistic	0,3579	0,4239	0,5289
Reference	0,4092	0,4657	0,5501

Table 1. Results for different teaching strategies received after adaptive phase – learning process is over. Grades [0-1].

Starting phase. All the students from the test groups were treated by the start phase strategies, and the experiment data were evaluated against optimal teaching criteria postulate – namely striving after receiving top grades with most possible credits earned and to match difficulty factor with student's expertise level.

Primitive strategy did not prove its usefulness, failing to match second part of requirements (differentiate students base on their expertise level). This strategy was not intended to be applied ever, in any system, and the results received after primitive strategy application, constituted a base for comparison. *Random* strategy, in spite of the fact that partially (since only some of the students were assigned lesson's difficulty factors that match their expertise level) met the requirements, gave good results (mainly in the group where most of the students were good learners - no matter what lesson's difficulty they were faced to they were able to cope with).

Proportional lesson difficulty variant assignment done by a proportional third strategy turned to be most efficient (against each test group), both in terms of average grades scored and credits earned. Thanks to it, more than 60% of students received better scores than expected.

Adaptive phase. Best strategy assessed in this phase should meet criteria described in section 4. In order to quickly sum up discussion of received results, if the priority was to get most students that passed learning path with higher than a group credit points average number, one should focus on Reference strategy or Optimal one. Focusing only average of grades maximization the most suitable are reference *conservative* and *reference* ones. Most balanced strategy that matches all criteria is *optimal* strategy. It equally good strives after grades and credits scored during entire learning path.

7. Conclusion

Quick and unfortunately chaotic e-learning systems development, created an urgent demand to adjust teaching process to the individual characteristics. Along with the growth of interests around distant learning, numerous systems are being implemented, yet again without orientation on a learner. The systems do not base on any student model or what is even worse do not adjust pace of learning to the student needs. This paper was intended to provide a solution to ease teaching material delivery, in an personalized way, that match student's expertise level. Based on a defined model of learner - system has to ascribe a teaching strategy that will facilitate knowledge acquisition during entire learning process. Tested, in two phases (start and adaptive) strategies allowed to increase learning effectiveness, portrayed by the results (increased number of credit points and average of grades) gathered in the table 1.

Future work will encompass experiments with real students as a part of working production e-learning system.

8. References

- Al-Dahoud A., Walkowiak T., Woda M. (2008). Dependability aspects of e-learning systems. *Proceedings of International Conference on Dependability of Computer Systems, DepCoS - RELCOMEX 2008*, pp. 73-79, Szklarska Poręba, June 2008, IEEE Computer Society [Press], Los Alamitos
- Baloian N., Motelet O., Pino J. (2003) Collaborative Authoring, Use and Reuse of Learning Material in a Computer-integrated Classroom, *Proceedings of the CRIGW 2003*, 2003, France.

- Bacopo A. (2004) Shaping Learning Adaptive Technologies for Teachers: a Proposal for an Adaptive Learning Management System, *Proceedings of the 4th IEEE International Conference on Advanced Learning Technologies*, 2004.
- Capusano N., Marsella M., Salerno S. (2000) An agent based Intelligent Tutoring System for distance learning, *Proceedings of the International Workshop on Adaptive and Intelligent Web-Based Education Systems*, ITS, 2000
- Dinosoreanu M., Salomie I. (2003) Mobile Agent Solutions for Student Assessment in Virtual Learning Environments, *Proceedings of the IAWTIC*, 2003, Austria
- Kavcic A. (2000) The role of user models in adaptive hypermedia systems, *Proceedings of the Electrotechnical Conference MELECON*, 2000.
- Mghawish A., Woda M., Michalec P. (2006). Computer aided composing learning material into primary learning tree. *Proceedings of the WSEAS International Conferences: The 5th WSEAS International Conference on Applied Computer Science (ACOS '06)* pp. 80-85, Hangzhou, April 2006, WSEAS 2006 Hangzhou.
- Nichols, M. (2008). E-Learning in context. <http://akoaootearoa.ac.nz/sites/default/files/ng/group-661/n877-1---e-learning-in-context.pdf>
- Woda M. (2008). Zarządzanie procesem uczenia w komputerowych systemach wspomagających nauczanie. *Proceedings of the Nowe Media w Edukacji 2008: zastosowania technik informacyjnych i komunikacyjnych w kształceniu*. pp. 147-155, Wrocław, September 2008, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław
- Woda M., Walkowiak T. (2008). Wybór optymalnej platformy zdalnego nauczania. *Proceedings of Komputerowe wspomaganie dydaktyki : materiały krajowej konferencji naukowej*. pp. 119-122, Łódź, June 2008, Wyższa Szkoła Informatyki, Łódź
- Woda M. (2006). Conception of composing learning content into learning tree to ensure reliability of learning material. *Proceedings of International Conference on Dependability of Computer Systems. DepCoS - RELCOMEX 2006*, pp. 374-381, Szklarska Poręba, May 2006, IEEE Computer Society [Press], Los Alamitos
- Woda M., Walkowiak T. (2004). Internet - a modern e-learning medium. *Proceedings of the Second International Conference on Soft Computing Applied in Computer and Economic Environments ICSC 2004*. pp. 205-214. Kunovice, Czech Republic, January 2004. Evropsky Polytechnicky Institut, Kunovice.

Measuring Customer Service Satisfactions Using Fuzzy Artificial Neural Network with Two-phase Genetic Algorithm

M. Reza Mashinchi and Ali Selamat

*Faculty of Computer Science and Information System, Universiti Teknologi
Malaysia*

Abstract

In this chapter, we propose a new method based on genetic algorithms (GAs) for fuzzy artificial neural network (FANN) learning to improve its accuracy in measuring customer service satisfaction for establishing a principle of economical survival in business area. The analysis is based on linguistic values received from customer service satisfactions index where fuzzy modeling, as one of possible ways, has been used to process these values. Here, customer's satisfaction is considered as a key factor for the analysis based on his/her preference as the scope of qualification for organization service. In the proposed method, we have introduced two-phase GAs-based learning for FANNs. In the neural network, inputs and weights are assumed to be fuzzy numbers on the set of all real numbers. The optimization ability of GA is used to tune alpha-cuts boundaries of membership functions for fuzzy weights. Here, five alpha-cuts are used for tuning as other researchers have used, which in two-phase method; two of them are for first phase and three of them for second phase. This leads to obtain better results for FANN. Comparisons are included with another method using two data sets to give some analyses to show the superiority of proposed method in term of generated error and executed time. From the experiments, the proposed approach has been able to predict quality values of possible strategies according to customer's preference. Finally, the ability of this system in recognizing customer's preference has been tested using some new assumed services.

Key words: Weight adjusting; placement definition; shape definition.

1. Introduction

Selling rate of the products for an enterprise, either be a business centers or producer factories, is an important issue in the commercial competitions. The higher rate an enterprise gains the more merit for survival is proved. Earlier researches have shown that the increase or decrease of this rate highly depends on customers' view to that commercial enterprise [1,2]. Such that; the more ability of satisfying the customer an enterprise has, the more

success in competition with other competitors it will achieve. As customers' satisfaction plays a key role in an enterprise survival, the analysis of his/her opinion is vital to make the next enterprise decisions. In general, customer's satisfaction is not only a multi-variable issue but also is based on linguistic values. On the other hand, linguistic values, which have been used here, are intrinsically known as vague values [3]. The two mentioned multi-variability and linguistic-variability make the problem to be more complex and system evaluation would be more difficult. This is while; strategic goals of the enterprises are determined according to the results of this analysis, and thus, the evaluation of customer opinion is an essential issue [2,4,5]. A suitable evaluation significantly helps an enterprise to emerge its defined strategic goals. This needs to have a well understanding of customer's opinion in order to be able of approximating his/her satisfactory degree.

Customer's satisfaction is satisfactory degree of the customer, which he/she is purchasing commodities [4]. Some indicators measure this degree. The indicators and its parameter are non-standard, and thus, each enterprise has been established an index according to its own customer's view [4,5]. Some indices, which are well known among the others, are American Customer Satisfaction Index (ACSI), Swedish Customer Satisfaction Index (SCSI), European Customer Satisfaction Index (ECSI) and Korean Customer Satisfaction Index (KCSI) [4,5]. It is worth mentioning that the parameters of indicator must be visible to customers' view [5]. According to the literature, three basic aspects of independency, comparability and feasibility must be considered [1, 2, 4]. In this chapter, indices have been used that supports the mentioned aspects employed by other researchers. One analysis ways of utilized indices, which is based on linguistic values received from the customer, is fuzzy modeling [3]. It is used in many papers for evaluation of the customer's satisfaction in the e-commerce, where the data of this area is utilized in this chapter. Various methods have been considered based on this modeling. Some researches have been used AHP [6-8] or either fuzzy cognitive maps [9]. Recently, some literatures have been appeared based on the combination of linguistic variables modeled by triangular fuzzy values [4,5].

Following aforementioned researches, this chapter aims to propose an evaluator system that would be able to recognize customer's preference. Meanwhile, it uses the benefits of fuzzy modeling in customer satisfactory evaluation. This mentioned aim of constructing a system that recognizes customer's preference has not been considered by the other authors. This is the difference of this research with the others'. In order to construct such a system proposing an approach, as the major part of the system, that can consider two terms of the learning and linguistic values is essential. This approach needs to follow learning process based on linguistic values in addition of having the ability to learn from customer's opinions. This task is carried out using Fuzzy Artificial Neural Networks (FANNs), which are know as soft computing techniques. FANNs are able to learn from fuzzy values that are considered as linguistic terms. Meanwhile, the Genetic Algorithm (GA) has been used to obtain higher efficiency for FANN. GA is able to find the optimum of designed network. Finally, each enterprise will be able to have the benefits of using such constructed system as follows:

- To evaluate possible strategies according to its current customer's tastes, in order to increase success rate;
- To analyze a strategy, regardless its business level, using the least number of the customers;

- To decrease the risk exists behind the decision making for its next organizational changes;
- To emerge the importance of business ethics, followed by customer-orientation principle.

It is necessary to have an evaluator system with a higher accuracy to decrease strategic decision risks and increase the success rate by evaluating possible strategies based on the preferences of the least customers. Such that; the higher accuracy of the system there is, the better organizational changes are obtained. Thus, it increases the success rate while emerges the business ethics. Knowing that an enterprise needs organizational changes to be adaptable with customer preferences, having a higher accuracy system rises to be necessity as listed below:

- to have more effective participation from customer side;
- to have more participant customers;
- to decrease the computational costs of evaluator system

Each enterprise needs to consider more effectively customers' opinion to have better understanding of their preferences, and thus, having a precise data is necessary for organizational evaluation. This is while; having variety opinions are necessary in covering broader preferences to have an assured organizational evaluation. Thus, the evaluator system is able of better approximation for new possible preferences. However, obtaining to points 1 and 2 comes with increase of the complexity, such that; the increase of data preciseness causes less accuracy for evaluator system, while the increase in number of data being processed causes less accuracy too. Therefore, a system that is able of dealing with such data environment is necessary. However, the ideal is such an evaluator system that is able to increase the accuracy while complexity increases. Such system, which this chapter proposes, enables an enterprise to have more assured evaluation in less time.

Based on previous research [10], this chapter follows to improve the last results. Therefore, the aim of proposed system is to improve the previous one using two phases for evaluator system. This system, called two- phased GA-based FANN, utilizes the abilities of GA to find a suitable status of evaluator system in order to improve the accuracy [11]. The first and second phases are called place-definition and shape-definition, respectively. Alpha-cuts (α -cuts), which here the fuzzy numbers are processed based on them, are defined and applied in each phase separately. The abilities of such evaluator, relative to complexity increase, on overcoming aforementioned complexities are:

- to decrease the predicted outcome error;
- to increase the processing speed;

Finally, using such system will enable an enterprise to:

- Have the ability of processing more realistic data received from the customer;
- Have more precise evaluation and suitable strategy approximation to increase the success rate in less time;

- Decrease the risks behind the organizational changes in terms of increasing preferences reality;
- Survive ethical principle of customer-orientation by customer participation.

The organization of this chapter, which aims at proposing such customer evaluator system using GA-based FANN, is as follows. First, in next section the concept of customer evaluator system has been explained. Then, how to model the current problem using fuzzy modeling has been explained in its first subsection. In the sequel, proposed evaluator systems using GA-based FANN and its basic concepts have been explained in its second subsection. Proposed system is implemented in following section and the results have been analyzed in its two subsections. In these subsections, first, the performance validation of proposed system has been tested using new inputs and then the ability of system has been shown. Finally, a conclusion for the chapter is given. Two datasets, which are used in this chapter, have been presented in appendix.

2. Customer Evaluator System

Customer evaluator system defines service quality of an organization based on customer's satisfactory degree [4]. The structure of such evaluator system is shown in Figure 1. The outcome of system is based on customer's opinion given to the system, which is represented by the parameters of some indicators. The preciseness of outcome exploration depends on the preciseness of modeled opinion expressed from the customer and the processing ability of evaluator system based on them.

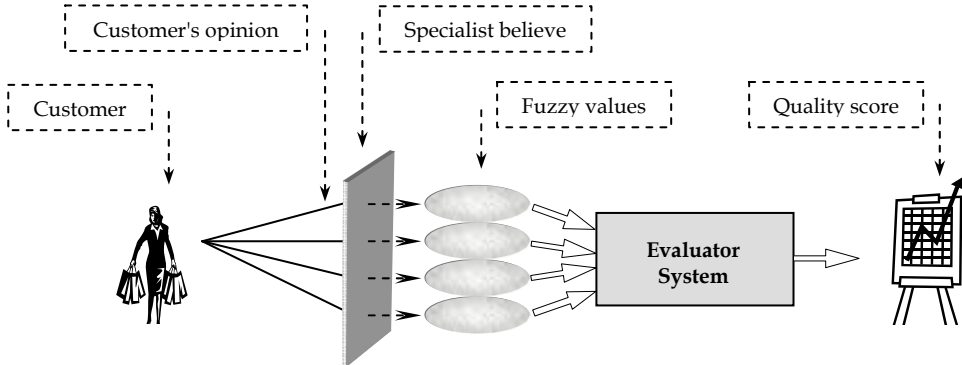


Fig. 1. The structure of the evaluator system

Hence, it is essential to have the parameters for evaluator system. They are presented by some indicators indices so-called customer satisfactory indices [4,5]. As aforementioned, these indices are not standard and this chapter uses the one that has been employed in [4]. The parameters of these indices are shown in Figure 2.

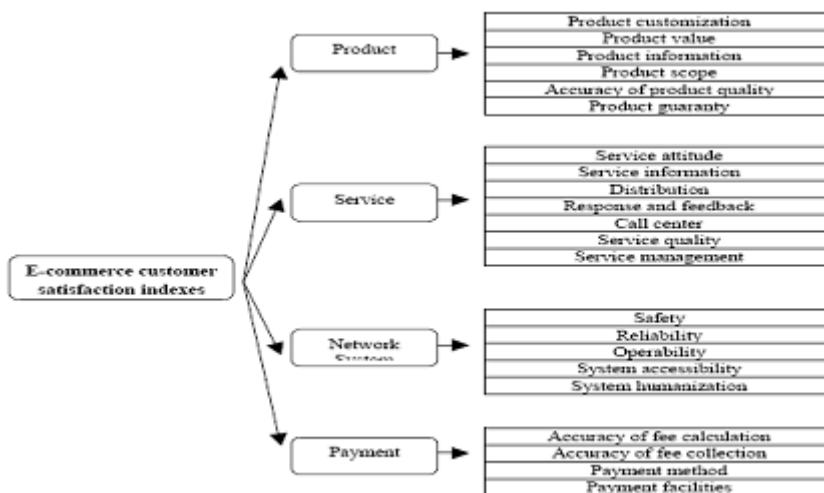


Fig. 2. A customer satisfactory indices and its parameters

The parameters in Figure 2 are represented in the form of linguistic variables that are accounted as vague values [3]. As these parameters are the inputs of system, they are necessary to be prepared for processing. To this end, fuzzy modeling is utilized in such way that first; the input values are fuzzified, then they are fed to FANN system to be processed. More details of these procedures are described in the following subsections, respectively.

Fuzzy modeling: Fuzzy set theory, which was proposed in 1965 [12], is utilized in many application areas by solving their corresponding problems [13,14,15,16]. This is due to the ability of fuzzy logic, with its modeling capability, in facing with complex environments [17,18]. In such environments like agriculture, market prediction, risk assessment [19], image processing etc. [16] the linguistic variables can be used. Customer satisfactory evaluation, the dealt issue in this chapter, is among such environments; this is because the process of this evaluation is based on information steamed from linguistic terms. In addition, having many linguistic variables causes this problem to be as a multi-variable issue. The latter one makes the problem to be more complex and, thus, a suitable modeling is much more needed for a better problem solving. Therefore, in this chapter, fuzzy modeling is considered to process linguistic terms that are received from the customer in order to construct evaluator system. Fuzzy variables, which are used in the evaluator system, are modeled linguistic terms received from the customer. Modeling a fine linguistic term needs more α -cuts as the preciseness of a fuzzy value depends on them. On the other hand, increasing the number of α -cuts causes the evaluation process to be more complex. However, the proposed approach in this chapter, which has a particular view to fuzzy modeling, considers this issue solvable.

The evaluator system uses the fuzzy variables for the inputs and parameters of the evaluator neural network. It is noticeable for the fuzzy values utilized in the input that; the transformation of these values depends on experts' interpretation over linguistic terms,

which is done through specialists' believes as illustrated in Figure 1. However, the idea of this chapter is concentrated on evaluator system only and, thus; obtained fuzzy values have been used based on other researchers' results [5]. Here, five α -cuts are used for fuzzy valued parameters of evaluator system to explore fuzzy numbers in its evaluation process. To be self-contained, we quote some fuzzy arithmetic on fuzzy numbers where a fuzzy number, \tilde{A} , defined as below: $\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in \mathfrak{R}, \mu_{\tilde{A}} : \mathfrak{R} \rightarrow [0,1]\}$

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in \mathfrak{R}, \mu_{\tilde{A}} : \mathfrak{R} \rightarrow [0,1]\} \quad (1)$$

such that $\mu_{\tilde{A}}$ is a continues function and \mathfrak{R} is set of all real numbers. A-cuts zero, one and middle of fuzzy number \tilde{A} are deified as below:

$$\tilde{A}_1 = Core(\tilde{A}) = \{x \in \mathfrak{R} \mid \mu_{\tilde{A}}(x) = 1\}, \quad (2)$$

$$\tilde{A}_0 = Support(\tilde{A}) = \{x \in \mathfrak{R} \mid \mu_{\tilde{A}}(x) > 0\}, \quad (3)$$

$$\tilde{A}_\alpha = middle_\alpha(\tilde{A}) = \{x \in \mathfrak{R} \mid \mu_{\tilde{A}}(x) \geq \alpha\}, \alpha \in (0,1). \quad (4)$$

Henceforth, all fuzzy numbers are assumed convex, such that all $middle_\alpha(.)$ are intervals in \mathfrak{R} and their $Support(.)$ are bounded. Two basic operations of the summation and multiplication over triangular fuzzy numbers, which are used in proposed evaluator system, are defined as follows [18]:

$$\begin{aligned} \tilde{A}_\alpha(k) + \tilde{B}_\alpha(k) &\equiv [\tilde{A}_\alpha^L(k), \tilde{A}_\alpha^R(k)] + [\tilde{B}_\alpha^L(k), \tilde{B}_\alpha^R(k)] \\ &= [\tilde{A}_\alpha^L(k) + \tilde{B}_\alpha^L(k), \tilde{A}_\alpha^R(k) + \tilde{B}_\alpha^R(k)] \end{aligned}$$

$$\begin{aligned} \tilde{A}_\alpha(k) \cdot \tilde{B}_\alpha(k) &\equiv [\tilde{A}_\alpha^L(k), \tilde{A}_\alpha^R(k)] \cdot [\tilde{B}_\alpha^L(k), \tilde{B}_\alpha^R(k)] \\ &= [\min(\tilde{A}_\alpha^L(k) \cdot \tilde{B}_\alpha^L(k), \tilde{A}_\alpha^L(k) \cdot \tilde{B}_\alpha^R(k)), \max(\tilde{A}_\alpha^R(k) \cdot \tilde{B}_\alpha^L(k), \tilde{A}_\alpha^R(k) \cdot \tilde{B}_\alpha^R(k))] \end{aligned}$$

Evaluator neural network: Neural learning networks are among soft computing techniques [20]. Learning networks of the fuzzy-type, so-called FANNs, were proposed after crisp neural networks [21]. FANNs have attracted many researchers in consequent of acquiring improvements and knowing their abilities in solving the complex problems. In this chapter, FANNs have been used as the main part of customer evaluator system. Evaluation outcome is processing result from the system based fed inputs. This network is able to learn customer's preference by its training process. Training process is based on the fuzzy values resulted from linguistic value transformations. Therefore, trained networks will be able to predict customer's satisfactory degree based on current preference, such that; it allows to

approximate the goodness of new organizational changes to be applied. The steps of constructing such system, as a general case of customer evaluator neural network, are explained in Algorithm 1.

Algorithm 1: The steps of the evaluator neural network

```
1 begin
2   initialize ()
3   x ← create ()
4   while ¬ terminationCriterion() do
5     xnew ← update (x)
6     if f(xnew) < f(x) then x ← xnew
7   return x
8 end
```

Step 2 of Algorithm 1 initiates the indicators with fuzzy values. Then, it creates possible solutions by aiming to find a suitable network and x will be replaced with that. Reproduction process, as the update function and finding the better solution, is repeated until it meets the termination criterion. A criterion for a near-optimal solution is;

$$\|f(x_{\text{new}}) - f(x)\| < \varepsilon \quad (5)$$

where f is fitness function, $\|\cdot\|$ is distance norm and ε is a given pre-assumed positive small number as error bound.

Model structure of Fuzzy Artificial Neural Networks (FANN): This subsection presents the structure of FANN [22]. Here, FANN of type-1 is used as the major part of evaluator system, in which the input is fuzzy value and the output is crisp [17,23,22,11]. Input neurons have been used to learn customer's preference based on the α -cuts defined in [3]. The structure of such FANN, using two inputs a general architecture, is shown in Figure 3.

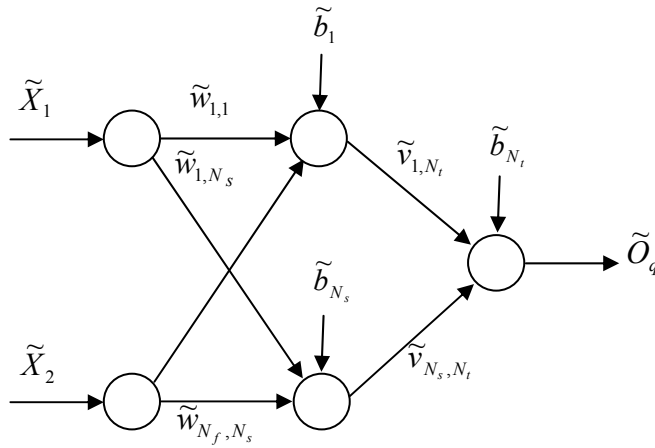


Fig. 3. A three-layer fuzzy neural network architecture

The first layer is input layer which does not have any computational unit. In the second layer, the matrix of fuzzy weights, \tilde{w}_{N_f, N_s} , shows fuzzy weights connecting neuron N_f in the first layer to neuron N_s in the second layer. The vector of fuzzy biases, \tilde{b}_{N_s} , shows fuzzy bias of the neuron, N_s , in the second layer. Similarly, in the third layer, fuzzy weight matrix, \tilde{v}_{N_s, N_i} , shows fuzzy weights connecting neuron N_s in the second layer to the neuron N_i in the third layer. The form of activation function of the neurons in the first and second layers, which is utilized in this chapter, is sigmoid function given as below:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (6)$$

Fuzzy output of $Int\tilde{er}_{N_s}$, for the second layer of this architecture is as follows:

$$Int\tilde{er}_{N_s} = f(A\tilde{g}g_{N_s}), \quad N_s = 1, 2, \dots, n \quad (7)$$

where N_s is the number of neurons in the second layer and $A\tilde{g}g_{N_s}$ is defined as follows:

$$A\tilde{g}g_{N_s} = \sum_{i=1}^{N_f} \tilde{X}_i \cdot \tilde{w}_{ij} + \tilde{b}_j, \quad j = 1, 2, \dots, N_s \quad (8)$$

where N_f is the number of neurons in the first layer and \tilde{X} is fuzzy input. The third layer receives the $A\tilde{g}g$ values from the neurons in second layer through their fuzzy weight \tilde{v} . Therefore, the output is given by:

$$\tilde{O}_q = \sum_{j=1}^{N_s} A \tilde{g}_j \tilde{v}_{jq}, \quad q = 1, 2, \dots, N_t \tag{9}$$

where N_t is the number of neurons in the third layer and \tilde{O}_q is fuzzy outcome. Then, the outcome is considered as a crisp value when the distance is measured with the ideal.

Genetic algorithm based FANN: Genetic algorithm (GA) was first proposed in 1975 [24]. It is categorized as an optimization and soft computing technique, which is based on the principles of natural evolution [20,21]. Here, optimization process holds on defined generations, where GA is used toward improving the efficiency of FANN. The idea of using GA for improving FANN was first proposed in 1994 [21].

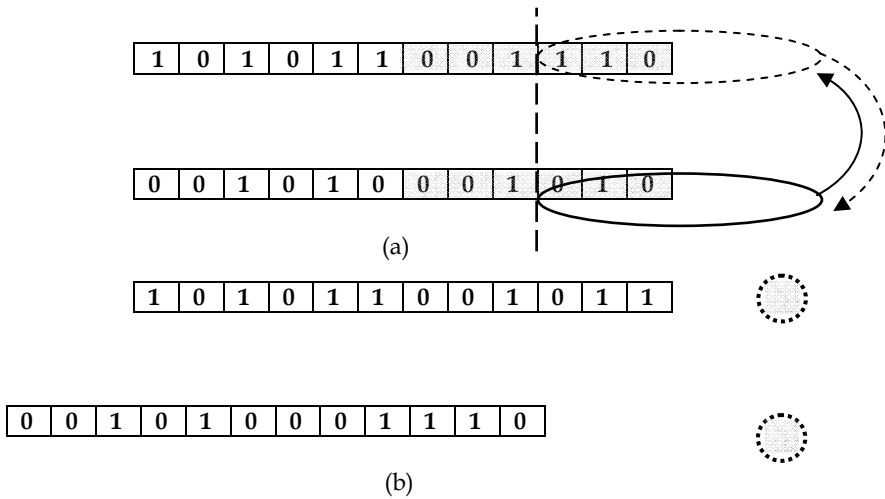


Fig. 4. (a) A typical crossover, (b) A typical mutation

Here, this idea, called GA-based FANN, is used as a validated method [25]. In GA-based FANN, genetic algorithm tries to find network parameters in an optimized manner. Such that tuning the weights and biases is an aim to find suitable network through optimization process of this algorithm. To this end, firstly, the parameters of network are simulated as the genes on a genome, then crossover and mutation functions, as a reproduction process, follow optimization process in an evolutionary way. These two functions have been illustrated as in Figure 4. Depicted (a) of this figure illustrates crossover function in which the gene of segmented parts of genomes are replaced. The mutation, which holds after the crossover in GA optimization process, has been illustrated in part (b) of this figure; in which the values of some defined genes are changed randomly. In this chapter, an improved case of previously used GA-based fuzzy neural network in last research is used as major part of the system as shown in Figure 5 [11]. In comparison, the previous system was using one phase GA-based FANN abbreviated as 1P-GBLM-ES; while this system uses two phase GA-based FANN abbreviated as 2P-GBLM-ES shown in Figure 6.

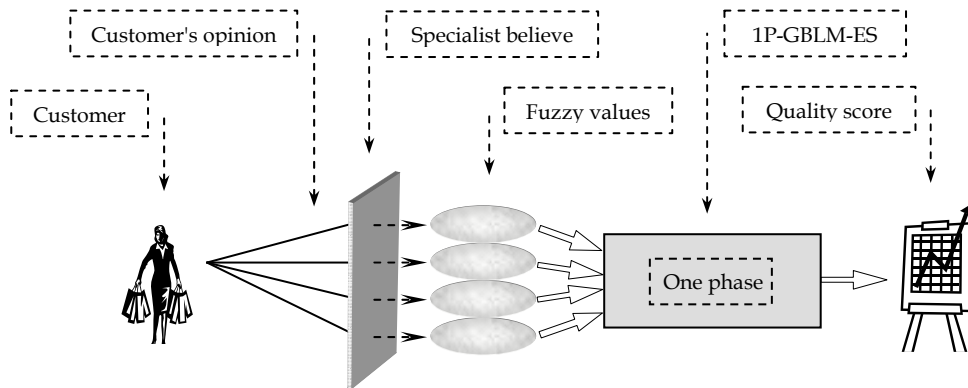


Fig. 5. general structure of 1P-GBLM-ES

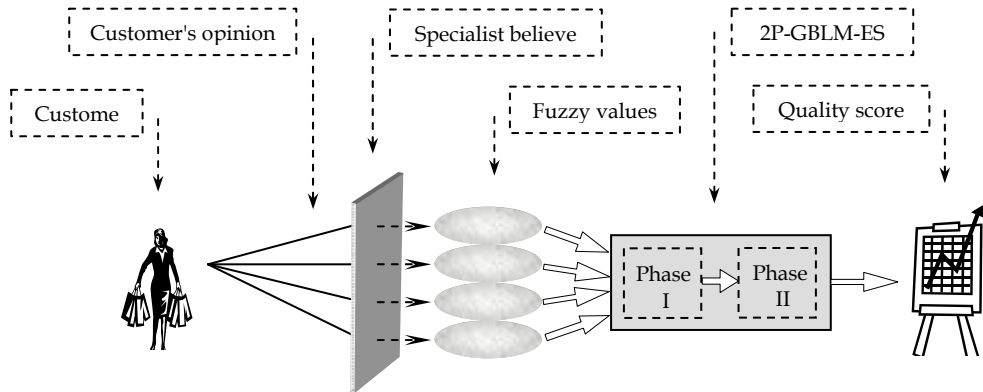


Fig. 6. general structure of 2P-GBLM-ES

Figure 5 explains the steps of constructing 2P-GBLM-ES. This system is designed in two phases of place-definition and shape-definition; in which a suitable place for fuzzy value is obtained in steps two to eight using support and core of the fuzzy numbers and a suitable shape using $middle_e$ in steps nine to sixteen [11].

Algorithm 2: The steps of the 2P-GBLM-ES

```

1  begin 2P – GBLM
2    begin phase I
3      initialize ( )
3       $x_1 \leftarrow \text{create} ( )$ 
4      while  $\neg \text{terminationCriterion} ( )$  do
5         $x_{new} \leftarrow \text{update} (x_1)$ 
6        if  $f(x_{new}) < f(x)$  then  $x_1 \leftarrow x_{new}$ 
7        return  $x_1$ 
8    end phase I
9    begin phase II
10     initialize using  $x_1$  boundaries ( )
11      $x_2 \leftarrow \text{create} ( )$ 
12     while  $\neg \text{terminationCriterion} ( )$  do
13        $x_{new} \leftarrow \text{update} (x_2)$ 
14       if  $f(x_{new}) < f(x)$  then  $x_2 \leftarrow x_{new}$ 
15       return  $x_2$ 
16    end phase II
17 end 2P – GBLM

```

Here in the implementations, heuristic and uniform models have been used for the crossover and mutation, respectively. It is noticeable that; here GA as an optimization technique is a lateral part of major FANN in evaluator system.

3. Implementation And Results

Earlier, an approach were proposed to construct a customer satisfactory evaluator system based on the frame of Figure 6 using Algorithm 2. In this section, proposed approach is implemented to construct this evaluator system; a three-layer GA-based FANN with seven neurons are used to construct 2P-GBLM-ES. In order to have a higher accuracy for 2P-GBLM-ES, suitable allotments of learning generations for first and second phases are found. To emerge the superiority of proposed system in comparisons, another customer satisfactory evaluator system has been constructed based on the frame of Figure 5 using Algorithm 1. In this order, a three-layer GA-based FANN consisting of seven neurons have been used to construct 1P-GBLM-ES. The architecture of the networks are designed such that considers the simplicity and less complexity for the network. Two datasets are used to test the implementation; utilized datasets have been generated based on indicators data of [5], where customer's opinions have been shown based on pre-assumed indicators. Then, computed gap has been computed as the difference between expected value and satisfaction

value of customer's opinion from current status. The comparisons results for 1P-GBLM and 2P-GBLM are in terms of generated error and executed time; initially, the first subsection shows the validity of proposed approach by comparison, then the ability of proposed evaluator system has been shown in the second subsection using dataset of Table 4.

Validation of Proposed Evaluator System: This subsection shows performance validity of constructed 1P-GBLM-ES and 2P-GVLM-ES for approximating the gap in a small-scaled data environment using the dataset of Table 3 in Appendix. The results are supposed to be a direction of showing the capability for these systems in a more complex environments to show the superiority for 2P-GBLM-ES, which is presented in the next subsection. Regarding the validation test, two customer's opinions are considered that are almost in contradiction to each other as shown in Figure 7 based on Table 3. Then, neural evaluator system is trained to approximate the gap based on new data. The learning processes were done in the same conditions for initial population using size 50 for 200 generations, while the average of received errors were obtained in terms of 100 times iteration. In order to construct 2P-GBLM-ES, suitable allotments of learning generations for first and second phases were obtained as shown in Figure 10. Then, trained networks were tested for the validation; two new customers who had a middle opinion were evaluated, as shown in Table 1. One of the customers has an exactly middle opinion, where the gap resulted from his/her is exactly 5.1. The other customer has almost a fair opinion, where the gap resulted from his/her is a value around 5.1.

Regarding the validation for 1P-GBLM-ES, the average of generated error for trained system was 0.03 in 16-second time. The results of approximation for this training were obtained as shown in figures 8 and 9 for first and second test customers, respectively.

Customer \ Indicator	#1	#2
Product	(4.55,4.67,4.76,4.89,5.1,5.3,5.44,5.53,5.65)	(4.4,21,4.4,4.61,5.5,37,5.6,5.77,6)
Service	(4.5,4.61,4.74,4.85,5.1,5.35,5.46,5.58,5.7)	(4.4,28,4.54,4.8,5.3,5.77,6.02,6.23,6.5)
Network	(3.6,3.91,4.2,4.5,5.1,5.7,5.99,6.29,6.6)	(3.6,4.02,4.43,4.88,5.6,6.04,6.19,6.32,6.5)
System	(3.9,4.15,4.39,4.62,5.1,5.59,5.81,6.04,6.3)	(4.4,09,4.16,4.24,4.5,4.96,5.25,5.51,5.8)
Expected Gap	5.1	≈ 5.1

Table 1. Fuzzy values of the indicators for validation test.

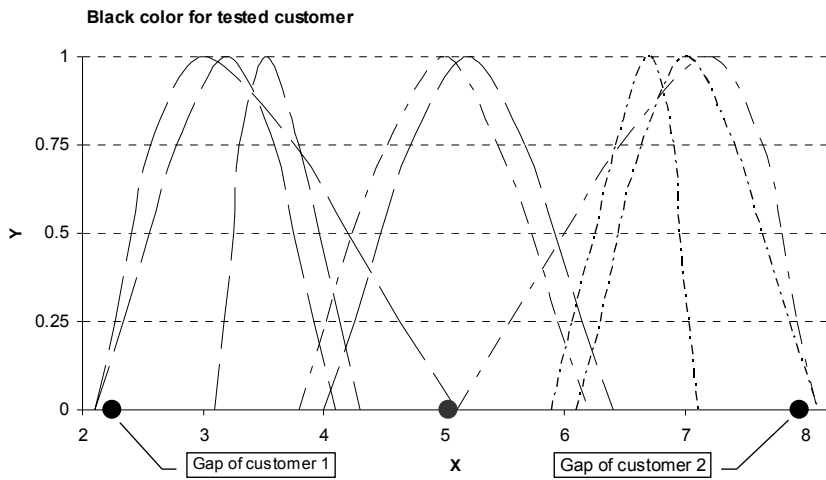


Fig. 7. Depicted opinions of the customers trained by evaluator systems

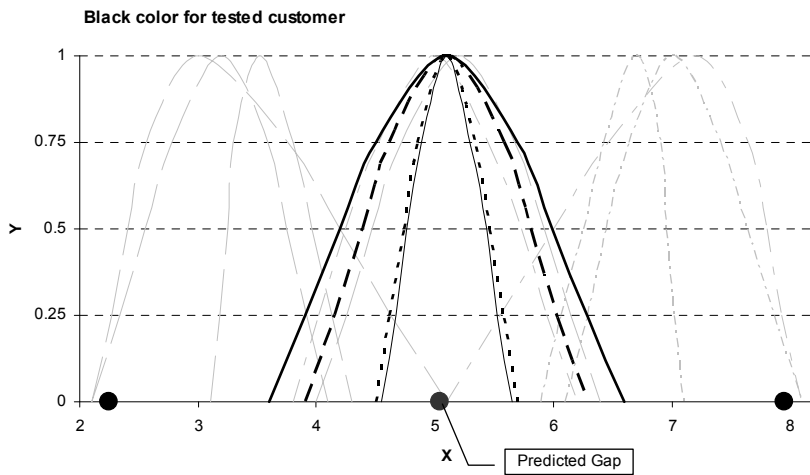


Fig. 8. Predicted gap for the first test customer using trained 1P-GBLM-ES

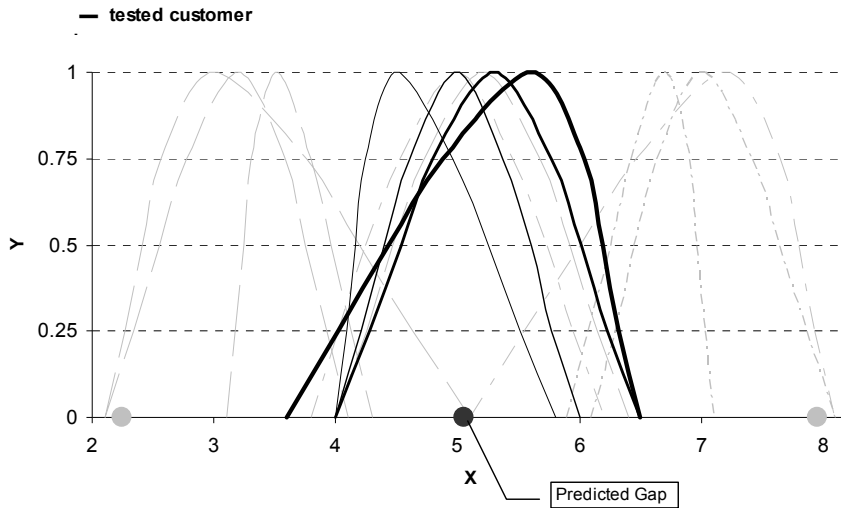


Fig. 9. Predicted gap for the second test customer using trained 1P-GBLM-ES

Figures 8 and 9 shows the values obtained from evaluating new assumed customers are 5.04 and 5.05, which are as expected. Therefore, the results show that customer-preference orientation of 1P-GBLM-ES in approximating customer's opinion works properly for a small-scaled environment. Then, regarding the validation for 2P-GBLM-ES, the same initial population was used to train the system as for 1P-GBLM-ES. In order to have a suitable accuracy for this system, different cases of allotments were tested to find a suitable case for each phase of the system. The result in Figure 10, which is the average of 100 iteration, shows the best-case belonging to 90% and 10% for the first and second phases, respectively. Then, 2P-GBLM-ES was trained using best-case allotment, where the average of overall generated error for the trained system obtained 0.002 in 11-second time. Figures 11 and 12 shows the outcome received from trained 2P-GBLM-ES using best-case allotment in evaluating new assumed customers that is 5.1 and 5.25 as expected.

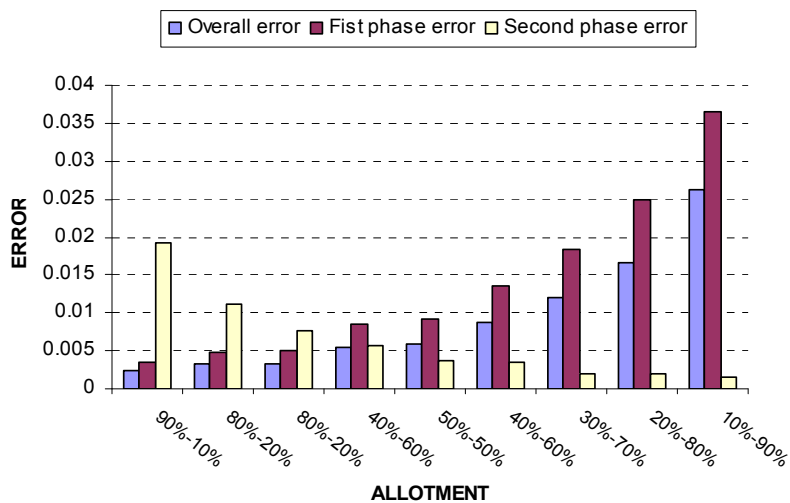


Fig. 10. Different cases of allotments for first and second phases of 2P-GBLM-ES using dataset of Table 3.

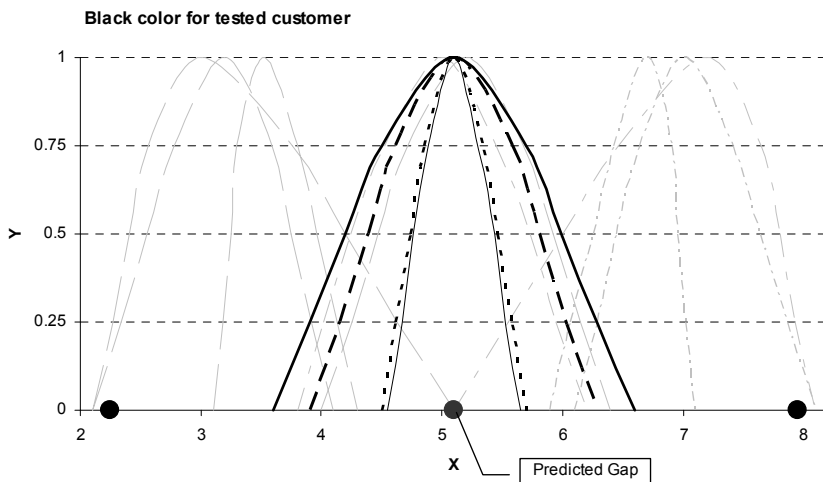


Fig. 11. Predicted gap for the first test customer using the trained 2P-GBLM-ES

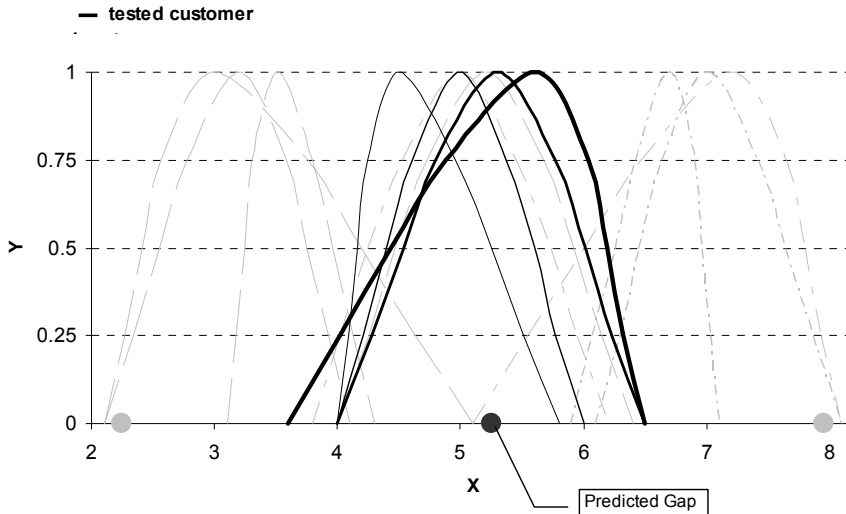


Fig. 12. Predicted gap for the second test customer using the trained 2P-GBLM-ES

The obtained results of figures 8, 9, 11 and 12 in this subsection, validate the performance of both 1P-GBLM-ES and 2P-GBLM-ES in approximating the gap. However, 2P-GBLM-ES showed to have less error in a less execution time.

Ability of evaluator system: In this subsection, the ability of 1P-GBLM-ES and 2P-GBLM-ES are analyzed in approximating the gap based on customer's opinion, in which more of them are exist in a more complex environment. The importance of this analysis is to show the approximation ability of both evaluator systems which are based on more available preferences. This makes resulted outcome to be more reliable for the enterprise and, thus, more assured strategic decisions could be taken. Here, ten customers data are utilized, which have been shown in Table 4 of Appendix. Learning behaviors are analyzed based on variation percentage of error decrease using different populations, in order to show the ability of constructed systems in learning data.

The results of learning process for 1P-GBLM-ES and 2P-GVLM-ES, using 500 generation and three different population sizes of 50, 200 and 500, have been shown in figures 13 and 15. Figure 11, which illustrates the learning behavior of 1P-GBLM-ES, shows that the variation rate of generated error keeps to its stability before the last generation just after achieving to its high value; this happens by passing a high decrement. Starting point of this, which is a threshold of convergence decrement to the optimum, happens in the 51st generation at the 10% of the 500 training generations. Vertical dashed-line shows this threshold in Figure 11, which the general status of suitable evaluator system is found. However, suitable evaluator system is defined after the threshold that needs ability of learning method in finding precise values. According to achieved threshold, it is possible to say in a pessimistic way that the ability of constructed system in learning the data of Table 6 and utilized populations is in the interval $(0.0054-\epsilon < \text{error} < 0.0263 + \epsilon)$.

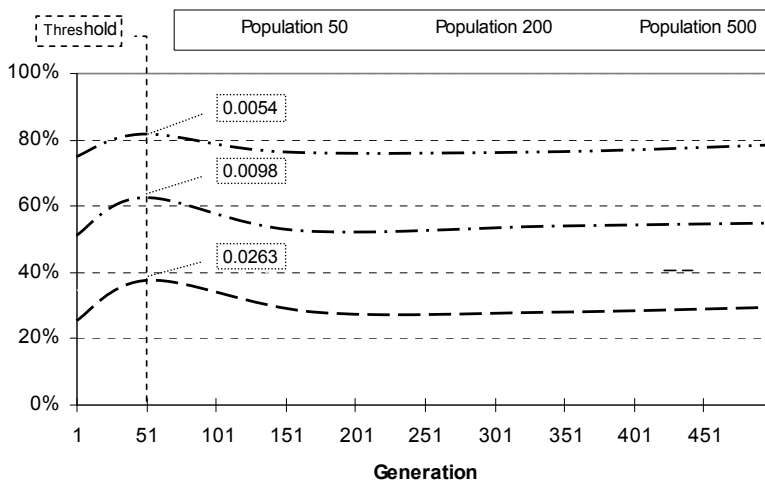


Fig. 13. The variation of generated error in different populations using 1P-GBLM-ES

Regarding 2P-GBLM-ES, different allotment cases of learning generations for its first and second phases was obtained using dataset of Table 6; so that the best-case allotment found in terms of generated error. Figure 12 shows the best-case allotment belonging to 90% of generations for first phase and 10% for second phase.

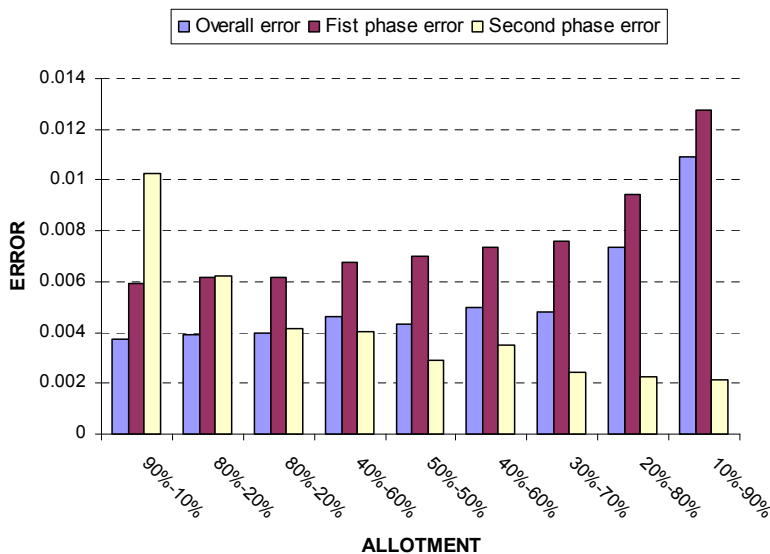


Fig. 14. Different cases of allotments for first and second phases of 2P-GBLM-ES using dataset of Table 4

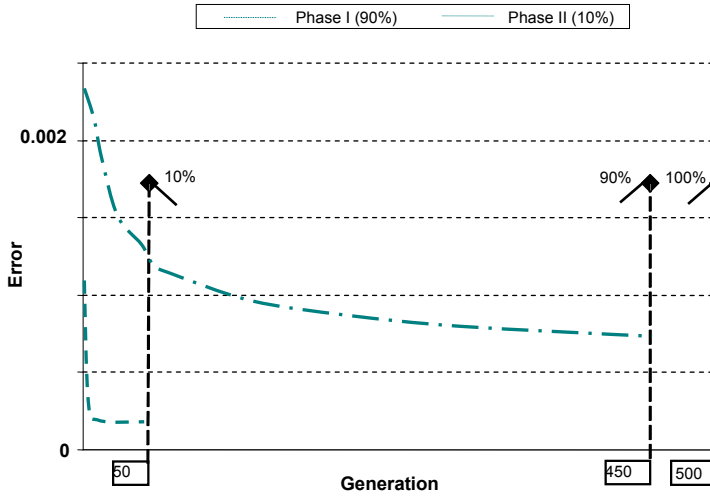


Fig. 15. The variation of generated error using 2P-GBLM-ES

Accordingly, Figure 15 shows the learning behavior of 2P-GBLM-ES using the based-case allotment. Here, the general suitability of the system is obtained in the first phase, where the second phase finds its precise status. Three customers have been used to test trained 1P-GBLM-ES and 2P-GBLM-ES based on Table 4. The suitable systems that has been used for this test is the case of 500 populations in 500 generations; while the error of 0.0263 for 1P-GBLM and 0.00371 for 2P-GBLM were obtained in 1518 and 1104 second-time respectively. Table 2 shows related outcomes received from obtained suitable evaluator systems based on the opinions of these customers. It is worth mentioning; the reason of using this case is the nature of dealt problem, which aims at strategic analysis for a higher quality approximation. Obtained results in this subsection, shows that both 1P-GBLM-ES and 2P-GBLM-ES has the ability of dealing with a precise customer data. However, 1P-GBLM-ES has less ability in the convergence, where it decreases after a threshold on learning generations. The threshold in Figure 13, showed that having the beneficial of 10% of learning generation, in which general suitable system is found, other remained 90% are most possible to be trapped in the local-minima. This shows disability of 1P-GBLM-ES in tuning the $middle_{\alpha}$, which is after 51st generation. This is while; 2P-GBLM-ES more avoiding to be trapped in the local-minima by separating the learning generations into two phases. Such that as the best-case allotment; 90% of learning generations are used to find the general suitability of the system, which was lost to be effectively used in 1P-GBLM-ES, and other remained 10% find precise status of the system. In addition, the separation of learning generations caused the speed of system to be faster in converging to suitable system. Therefore, 2P-GBLM-ES is superior in comparing with 1P-GBLM-ES in both terms of generated error and executed time.

4. Acknowledgment

This work is supported by the Ministry of Science & Technology and Innovation (MOSTI), Malaysia and Research Management Center, Universiti Teknologi Malaysia (UTM) under the Vot 79227.

5. Conclusion

This chapter considered an approach to facilitate a better analysis of strategic decisions to find a suitable strategy for a business enterprise; defining business strategy held on precise analysis of customers' opinions. It assumed customers' preferences as the major key in analysis, which is a new approach to solve the current problem. Regarding high complexity of customers' preferences in precise case, organizational analyze needed an approach being capable of obtaining an assured results. Therefore, this chapter proposed a system to enable an enterprise evaluating new possible organizational changes, in any business level. The superiority of this proposed system, which uses two phase genetic algorithm based fuzzy artificial neural network, was shown in comparison with one phase genetic algorithm based fuzzy artificial neural networks by some analysis. This was due to the ability of this system in finding general status of suitable evaluator system and more avoidance from trapping into the local-minima to find a precise status. This is while; this ability were based on fuzzy obtained best-case allotment for each phase of evaluator system. Finally, valid performance and the ability of system assure an enterprise to have beneficial consequences of using it for a risky condition in correct orientation as well as success rate in less time. However, its ability in dealing with more precise data and number of them may be found in a future research.

6. Appendix

Table 2. Test customers and predicted gaps using 1P-GBLM-ES and 2P-GBLM-ES

Customer Indicator	#1	#2	#3			
Product	(2.1,2.34,2.5,3.3,59,3.84,4.03,4.3,5.1)	(5.5,5.77,5.99,6.37,7.7,59,7.8,7.93,8.1)	(3.6,3.79,3.96,4.18,4.9,5.7,6.2,6.6,7.3)			
service	(3.1,3.28,3.48,3.76,4.5,4.74,5.07,5.36,5.78)	(6.7,6.8,6.83,6.97,7.3,7.49,7.57,7.58,7.6)	(3.7,4.27,4.87,5.48,6.4,6.42,6.44,6.49,6.6)			
Network	(2.3,2.35,2.41,2.44,2.6,2.69,4.2,4.6,5.1)	(4.8,5.77,6.13,6.71,8.0,8.05,8.08,8.09,8.1)	(2.6,3.23,3.67,4.27,5.6,5.88,6.04,6.3,6.5)			
System	(3.3,3.48,3.93,4.74,5.8,6.6,6.15,6.19,6.4)	(3.5,3.74,3.88,3.99,4.2,5.7,6.45,7.7,7.7)	(3.3,3.43,3.52,3.63,3.8,4.39,4.82,5.2,5.8)			
Gap/ System	1P-GBLM-ES 2.743	2P-GBLM-ES 3.625	1P-GBLM-ES 5.819	2P-GBLM-ES 6.758	1P-GBLM-ES 5.261	2P-GBLM-ES 3.205

Customer Indicator	#1	#2
Product	2.1,2.27,2.41,2.56,3.3,76,4.2,4.63,5.1	6.1,6.3,6.45,6.64,7.7,41,7.64,7.86,8.1
service	3.1,3.18,3.25,3.3,3.52,3.8,3.96,4.13,4.3	5.9,6.07,6.26,6.43,6.7,6.9,6.95,7.02,7.1
Network	2.1,2.34,2.55,2.78,3.2,3.59,3.75,3.92,4.1	5.1,5.55,6.6,46,7.2,7.65,7.8,7.93,8.1
System	4.4,26,4.47,4.72,5.2,5.67,5.92,6.15,6.4	3.8,4.02,4.24,4.53,5.5,45,5.73,5.94,6.2
Gap	2.25	7.95

Table 3. Dataset 1

Customer	#1	#2	#3
Indicator			
Product	(2,2,1,2,14,2,24,2,5,2,9,3,2,3,51,4)	(6,1,6,5,4,6,8,2,7,1,7,7,8,7,8,7,9,7,9,7,8,1)	(4,5,4,9,8,5,3,2,5,5,8,6,6,4,6,6,7,1,7,11,7,7)
service	(3,1,3,19,3,2,27,3,3,6,3,5,3,7,5,3,9,1,4,0,7,4,3)	(6,6,2,4,6,4,3,6,6,8,7,1,7,2,4,7,3,8,7,6,8)	(4,7,4,8,4,9,5,5,0,7,5,3,5,6,5,5,8,3,6,0,6,6,5)
Network	(1,9,1,9,1,9,1,9,2,3,2,6,2,9,5,3,5)	(7,7,2,4,7,5,7,7,8,8,2,8,2,8,2,8,2,8,2)	(5,5,5,9,6,1,3,6,3,2,6,7,7,0,8,7,2,7,7,6,8,1)
System	(4,4,3,1,4,5,6,4,8,1,5,2,5,7,3,5,9,4,5,1,5,6,4)	(3,8,3,9,4,0,7,4,3,7,5,5,3,5,5,8,5,9,1,6,2)	(2,4,2,7,1,2,9,4,3,2,3,7,4,0,3,4,2,8,4,6,3,5,2)
Gap	2	7,9	2,25
Customer	#4	#5	#6
Indicator			
Product	(2,2,0,8,2,1,5,2,1,7,2,3,2,8,9,3,1,6,3,5,9,4,1)	(3,7,5,0,2,5,7,2,6,4,7,3,7,4,7,7,5,8,7,7,7,8)	(4,8,5,5,1,6,5,3,7,5,8,6,1,9,6,5,7,7,0,1,7,7)
service	(2,3,2,6,1,2,8,1,3,0,5,3,4,3,4,6,3,5,3,3,5,9,3,7)	(6,6,6,0,9,6,1,6,6,4,6,7,2,7,0,9,7,4,9,8)	(2,6,2,9,2,3,4,8,3,9,4,2,5,4,4,4,4,7,5,1)
Network	(1,9,2,0,6,2,2,4,2,8,3,0,2,3,0,7,3,1,8,3,5)	(5,3,5,4,9,5,7,2,5,9,5,6,4,6,6,6,9,1,7,2,8,8,1)	(5,4,5,8,6,5,2,4,6,8,7,7,7,8,3,7,9,7,9,4,8)
System	(3,1,3,3,1,3,5,3,3,7,4,1,5,1,1,5,7,6,6,3,8,7,5)	(4,2,4,2,4,2,8,4,3,5,4,5,4,7,4,5,0,7,5,5,1,6,2)	(2,2,2,2,2,2,9,2,4,2,6,3,3,9,4,4,8,6,6,3)
Gap	3,36	7,01	5,91
Customer	#7	#8	#9
Indicator			
Product	(3,1,3,3,2,3,5,3,3,6,7,4,2,4,6,9,5,0,6,5,6,4,6,7)	(4,7,4,7,6,4,8,4,9,5,2,5,6,6,5,8,2,6,1,7,7)	(3,5,1,3,5,9,5,6,7,6,7,7,7,7,7,5,7,7,7,7,8,2,8)
service	(2,2,2,1,2,1,4,2,2,6,2,6,3,3,2,7,3,7,4,4,6)	(8,3,5,8,5,4,5,5,8,7,6,4,6,5,6,6,7,2,7,0,9,8)	(5,2,5,2,2,5,2,4,5,3,5,7,2,6,6,3,2,6,9)
Network	(2,8,3,7,4,4,3,7,4,9,7,6,6,2,8,6,3,6,6,4,5,6,7)	(6,6,8,6,7,2,3,7,5,3,8,8,8,8,8,1)	(2,6,3,0,1,3,1,5,3,4,3,9,4,3,2,4,4,8,4,7,6,5,3)
System	(2,6,2,7,4,2,8,1,2,9,6,3,3,3,6,3,8,4,4,0,7,4,5)	(5,7,6,1,6,2,5,6,5,6,7,7,0,4,7,0,7,7,0,7,7,2)	(4,2,4,2,8,4,3,4,4,4,4,7,5,3,8,5,8,2,6,3,4,7,7)
Gap	3,92	6,41	5

Table 4. Dataset 2

7. References

- [1] Zhao Pengxiang, 2001. Research on Building and Performance of Customer Satisfaction Management System. *World Standardization and Quality Management*, 6 (6): 10-13.
- [2] Zheng Yue-fang, 2005. Customer Satisfaction-The Most Important Task of Electricity Service. *China Quality*, 3: 65-66.
- [3] Hung, T. Nguyen and Elbert A Walker, 2006. *A First Course in Fuzzy Logic*. Third Edition, CRC Press London.
- [4] Peide Liu, 2007. Evaluation Model of Customer Satisfaction of B2C E-Commerce Based on Combination of Linguistic Variables and Fuzzy Triangular Numbers. In the Proceedings of the 8th IEEE International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, pp: 450-454.
- [5] Mehdi Fasanghari and Farzad Habibipour Roudsari, 2008. The Fuzzy Evaluation of E-Commerce Customer Satisfaction. *World Applied Sciences Journal*, 4 (2): 164-168.
- [6] Yi-wen Liu, Young-Jik Kwon and Byeong-do Kang, 2007. A Fuzzy AHP Approach to Evaluating E-commerce Websites. In the Proceedings of the 5th International IEEE Conference On Software Engineering Research, Management and Applications, pp: 114-124.
- [7] Mikhailov, L. and P. Tsvetinov, 2004. Evaluation of Services Using A Fuzzy Analytic Hierarchy Process. *Applied Soft Computing*, 5: 23-33.
- [8] Minghe Wang, Peide Liu and Guoli Ou, 2007. The Evaluation Study of Customer Satisfaction Based on Gray-AHP Method for B2C Electronic-Commerce Enterprise. *Engineering Letters*, 15 (1): 157-162.
- [9] Dimitris Kardaras and Bill Karakostas, 2006. E-service Adaptation Using Fuzzy Cognitive Maps. In the Proceedings of the 3rd International IEEE Conference on Intelligent Systems, pp: 227-230.
- [10] M. Reza Mashinchi, Ali Selamat, 2009, Constructing a Customer satisfactory Evaluator System Using GA-based Fuzzy Artificial Neural Networks, *World Applied Science Journal*, Vol. 5(4): 432-440, 2008.
- [11] M. Reza Mashinchi, Ali Selamat, 2009, An Improvement on Genetic-based Learning Method for Fuzzy Artificial Neural Networks, to appear in *Applied Soft Computing*.
- [12] Zadeh, L.A., 1965. Fuzzy sets. *Information and Control*, 8: 338-359.
- [13] Bellman, R.E. and L.A. Zadeh, 1970. Decision Making in A Fuzzy Environment. *Management Sciences*, 17: 141-164.
- [14] Kwang H. Lee, 2005. *First Course on Fuzzy Theory and Applications*. *Advances in Soft Computing*. Springer Berlin, Heidelberg.
- [15] Oscar Castillo, Patricia Melin, Oscar Montiel Ross, Roberto Sepúlveda Cruz, Witold Pedrycz and Janusz Kacprzyk, 2007. *Theoretical Advances and Applications of Fuzzy Logic and Soft Computing*. *Advances in Soft Computing*, Springer Berlin, Heidelberg.
- [16] Cengiz Kahraman, 2006. *Fuzzy Applications in Industrial Engineering Studies in Fuzziness and Soft Computing*. *Studies in Fuzziness and Soft Computing*, Springer Berlin, Heidelberg.

- [17] Hadi Mashinchi, M. and Siti Mariyam Shamsuddin, 2008. Three-term fuzzy Back_propagation. *Foundation on Computational Intelligence*, Book Springer, to be appeared.
- [18] Zadeh, L.A., 2005. Toward A Generalized Theory of Uncertainty (GTU)-An outline. *Information Sciences*, 172: 1-40.
- [19] Reza Mashinchi, M., M. Hadi Mashinchi and Ali Selamat, 2008. Wildfire Risk Assessment Using Fuzzy Artificial Neural Networks Estimation. In the Proceedings of the 4th International Colloquium on Signal Processing and its Applications, Kuala Lumpur, Malaysia, pp: 111-115.
- [20] Tsoukalas, L.H. and R.E. Uhrig, 1997. *Fuzzy and Neural Approaches in Engineering*. John Wiley and Sons Inc., New York.
- [21] Krishnamraju, P.V., J.J. Buckley, K.D. Reilly and Y. Hayashi, 1994. Genetic Learning Algorithms for Fuzzy Neural Nets. In the Proceedings of the 3rd IEEE Conference on Fuzzy Systems, IEEE World Congress on Computational Intelligence, pp: 1969-1974.
- [22] Liu, P. and H.X. Li, 2004. *Fuzzy Neural Network Theory and Application*. River edge, NJ: World Scientific.
- [23] Hadi Mashinchi, M., M. Reza Mashinchi, Siti Mariyam, H.J. Shamsuddin and Witold Pedrycz, 2007. Genetically Tuned Fuzzy Back-propagation Learning Method Based on Derivation of Min-max Function for Fuzzy Neural Networks. In the Proceedings of the International Conference on Genetic and Evolutionary Methods, Worldcomp Congress, USA, pp: 213-219.
- [24] Holland, J.H., 1975. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor.
- [25] Aliev, R.A., B. Fazlollahi and R.M. Vahidov, 2001. Genetic algorithm-based learning of fuzzy neural network. Part 1: feed-forward fuzzy neural networks. *Fuzzy Sets and Systems* 118: 351-358.

A Variation of Particle Swarm Optimization for Training of Artificial Neural Networks

Masood Zamani and Alireza Sadeghian
Ryerson University
Canada

1. Introduction

Particle swarm optimization (PSO) is a stochastic global optimization method (Eberhart & Kennedy, 1995) that belongs to the family of Swarm Intelligence and Artificial Life. Similar to artificial neural networks (ANN) and genetic algorithms (GA) which are the simplified models of the neural system and the natural selection of the evolutionary theory, PSO is a simplified model of psychological principles and social behaviors (Reynolds, 1987). PSO is based on the principles that flocks of birds, schools of fish, or swarm of bees searches for food sources where at the beginning the perfect location is not known. However, they eventually reach the best location of food source by means of communicating with each other.

PSO is also conceptually compared to evolutionary computation methods such as GA (Eberhart & Shi, 1998). The uniqueness of PSO is the dynamic interaction among the particles. The optimization method starts with randomly generated particles (the population) in a defined domain called the search space. Particle locations are updated in each generation (iteration) to explore the search space for an optimum solution. In PSO, particle positions and velocities are updated based on cooperation and competition. Each particle finds its next position in the search space according to its own search experience and the best experience of the particles located in its local group, neighborhood and the entire population. Neighbors are the particles located within a pre-defined distance of a specific particle.

In this article, we propose a method to update the velocities and positions of particles when the maximum search space boundary and velocity are reached. The efficiency of the proposed particle swarm optimization method is investigated through the training of feed-forward artificial neural networks used for classification. The experiments show the particle swarm optimization lends itself very well to training of neural networks and is also highly competitive with the other methods of training feed-forward ANNs. We have conducted four classification experiments using feed-forward ANNs with PSO based training. The data sets used in the experiments are from the UCI repository (Asuncion & Newman, 2007) commonly used in the literature.

2. Related works

The most widely used method of training for feed-forward ANNs is back-propagation (BP) algorithm (Hecht-Nelso R., 1989). Feed-forward ANNs are commonly used for function approximation and pattern classifications. Back-propagation algorithm and its variations such as QuickProp (Fahlman, 1998) and RProp (Riedmiller and Braun, 1993) are likely to reach local minima especially in case that the error surface is rugged. In addition, the efficiency of BP methods depends on the selection of appropriate learning parameters. The other training methods for feed-forward ANNs include those that are based on evolutionary computation and heuristic principles such as Genetic Algorithm (GA), and PSO.

Although, Genetic Algorithm (Mitchell M., 1988) is a suitable choice for the training due to its exploration and exploitation properties and solves the gradient-based drawbacks, however it suffers from the mutation problem leading to premature convergences and needs more time to converge to an optimum solution comparing to the particle swarm optimization. As we discuss about the properties of PSO later, it has been shown that PSO is a better evolutionary candidate for optimization (Eberhart, and Shi, 1998). The PSO algorithm possesses important characteristics such as memory and constructive cooperation among the individuals that can prevent mutation problem exist in GA. Different variations of PSO have been applied to train the feed-forward ANNs for non-linear function approximation and classification problems. The training of neural networks is achieved basically in two ways:

- 1- Adjusting the connection weights when the ANN structure is predefined such as the number of hidden layers, the number of neurons and their connections, and activation function parameters.
- 2- Evolving a ANN structure which is not predefined and adjusting the weights simultaneously.

Training of fixed structure ANN has been experimented by basic PSO method (Mendes et al., 2002). In this study, it has been shown PSO's performance is competitive to BP methods and especially in some problems where the number of local minima is high.

Also, the variant of PSO with minimum velocity constraint was proposed and tested for function approximations using feed-forward ANNs (Xiaorong et al., 2007). Applying the velocity constraint reduces the premature convergences and alleviates the effect of dimensionality increase. This is done by guiding the particle in the search space by limiting the maximum moving distance in each iteration. Thus, it prevents the particle to go out of the bound (search space) or to stop when the velocity increases or decreases. The very effective modifications focusing on optimizing the update equations of PSO were made in (Russ et al., 2000), (Kennedy, 2000). These modifications are adding the inertia weight and improving the PSO performance with cluster analysis. Using cluster analysis methods, the update equations are modified in a fashion that particle attempt to merge to the center of their cluster instead of merging to the global best location. This approach improves the performances in some classes of problems. In (Angelin, 1999), a selection mechanism was proposed for PSO similar to that already used in genetic algorithm to improve the quality of the particles in a

swarm. Another modified PSO is the cooperative learning proposed in (Van den Bergh & Engelbrecht, 2004). The application of this method to neural network training has yielded promising results. In this approach, input vectors are distributed into several sub-vectors which are optimized in their own swarms cooperatively. Performance improvement in this case is due to splitting the main vector into several sub-vector that in turn results in better credit assignments and reduces the chance to omit a possible good solution for a certain component in the vector.

Training of ANNs by Multi-Phase PSO (MPPSO) is another variation which evolves simultaneously multiple groups of particles that change the direction of search in different phases of the algorithm (Al-kazemi & Mohan, 2002). Each particle in this method is in a specific group and phase at a given time. MPPSO boosts the wider exploration of the search space, increases population diversity and prevents premature convergences. Furthermore, MPPSO has different update equations comparing to the basic PSO and permits changes to the locations of the particle that only lead to some improvements. PSO also has been used as a means to evolve ANN architectures (Chun-kai et al., 2000). In this study, the network structure is adaptively adjusted and the PSO algorithm is applied to evolve the nodes of the neural network with specific generated structure. The techniques such as the combination of partial training and evolving added nodes are employed to generate the desired architecture and then PSO is used to evolve the nodes of the pre-defined structure. Hybrid of genetic algorithm and particle swarm optimization (HGAPSO) is another modified PSO that was employed to design recurrent neural networks [Juang, 2004]. In HGAPSO method, the individuals of the next generation are created not only by crossover and mutation operators but also by PSO. The upper-half of the best-performing individuals in a population are enhanced using PSO and the other half is generated by applying the crossover and mutations. Unlike GA, HGAPSO removes the restrictions of evolving the individuals within the same generation. In this article, the proposed method is another variation of particle swarm optimization for fixed structure ANNs where only weights are adjusted.

3. Particle Swarm Optimization

The Particle Swarm Optimization algorithm is represented by the evolution of a population in the form of an n -dimensional vector $x=(x_1, \dots, x_n)$, $i=1, \dots, n$. These particles represent an approximation of the desired solution, and the number of dimensions depends on a given problem. Each particle has a memory p^i , $i=1, \dots, m$ where m is the number of particles) which keeps the best location that i^{th} particle has found since its search started. Furthermore, every flying particle has a velocity $v^i(t)$ that shows its direction and speed at the time instance t . In each iteration, particle locations and velocities are updated according to equations (1), and (2). The global best location, p^g , found by any particle, and local best location, p^{l_i} , found by neighbors of the i^{th} particle, are the two elements of shared information in the entire population. To evaluate each particle's performance, a fitness function is defined. There are two types of PSO, global and local (Bergh & Engelbrecht, 2002). The local version of PSO that is proven experimentally to be able to find the global optimum is shown by equation

(3). This method is computationally extensive since for each particle a neighborhood of size k is identified in each iteration as shown in Figure 1.

$$v^i(t+1)=w.v^i(t)+c_1.r_1(p^i-x^i(t))+c_2.r_2(p^g-x^i(t)) \quad , \quad i=1,\dots,m \quad (1)$$

$$x^i(t+1)=x^i(t)+v^i(t+1) \quad (2)$$

In the local version of PSO, the equation (1) is changed to (3).

$$v^i(t+1)=w.v^i(t)+c_1.r_1(p^i-x^i(t))+c_2.r_2(p^g-x^i(t))+c_3.r_3(p^{Li}-x^i(t)) \quad , \quad i=1,\dots,m \quad (3)$$

The initial values of positions and velocities are calculated for each particle by the equations (4), and (5).

$$v^i(0)=v_{\min} + \text{rand}(v_{\max}-v_{\min}), \quad i=1,\dots,m \quad (4)$$

$$x^i(0)=x_{\min} + \text{rand}(x_{\max}-x_{\min}), \quad i=1,\dots,m \quad (5)$$

In equation (1), r_1, r_2 are two random vectors with values ranging from zero to one. The inertia w is a predefined positive value that is decreased in each iteration to slow down the speed of particles which are closing gradually to the global best particle (Shi & Eberhart, 1998). As a result, this parameter gives more chance to particles to explore the search space and bound the increase of velocity. The expression $c_1.r_1(p^i-x^i(t))$ in equation (1) is the particle memory influence which indicates the scale that a particle relies on its own best past experience. Also, the expression $c_2.r_2(p^g-x^i(t))$ in equation (1) is swarm influence indicating the degree that a particle follows the best experience of the entire population or the local group which is shown the expression $c_3.r_3(p^{Li}-x^i(t))$ in equation (3). The three confidence measures which are self-confidence, swarm and local-group confidences are denoted by c_1, c_2 and c_3 respectively. These are positive constant values ranging from 1.5 to 2.5. The inertia value is usually chosen from 0.4 to 1.4 (Shi & Eberhart, 1998). Schematics for the equation (1), (2) are shown in Figure 2.

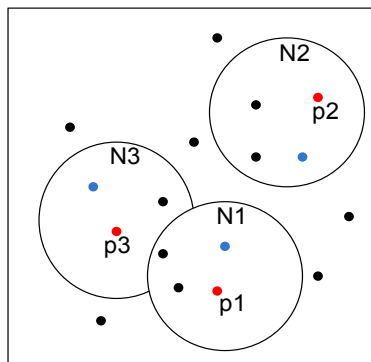


Fig. 1. Neighborhoods of size 4, N_i standing for neighborhood and P_i for particles

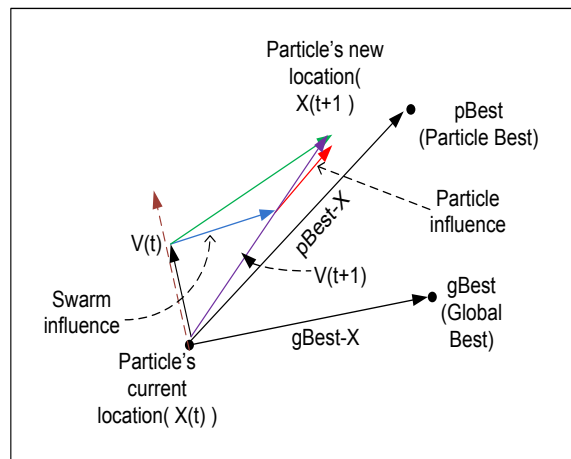


Fig. 2. The equation (1), (2) in the form of vector representation

The steps of PSO algorithm can be defined as the following:

- 1- Representing the primary solution of a given problem in the form of n-dimensional vectors.
- 2- Defining a boundary for the search space and maximum velocity.
- 3- Defining a fitness function to evaluate the quality of each particle.
- 4- Generating a population consist of m particles represented in the form of n-dimensional vectors and locating them randomly in the search space.
- 5- Updating the position and velocities of the particle by the equation (1),(2).
- 6- Evaluating the fitness of each particle
- 7- Updating the best experience or memory of each particle and the global best particle.
- 8- Repeating from the step 5 until the desired solution is achieved.

PSO is comparable to Evolutionary Computation (EC) methods namely genetic Algorithm (GA) since it is a stochastic population based method. In PSO the information is shared among the population by global best and local best particles whereas in GA the crossover operator performs the same function. The random vectors r_1, r_2 in the equation (1) also are similar to the mutation operator in the GA. However, unlike GA that discards the individuals with lower fitness, in PSO all particles are kept and transferred to the next generation. In addition, each particle has memory, whereas in GA the best individual is kept in each generation.

4. Methodology

Training of neural networks can generally be considered as modification of the randomly generated weights of pre-defined ANNs that comprise of a certain number of inputs, outputs, and neurons in different hidden layers. The weights are changed until the difference between the actual ANNs outputs corresponding to input (samples or training set) and the desired output reaches a certain error. Therefore, the training of ANNs can be

interpreted as constructing a model (function) and optimizing it in an n -dimensional space. That is, we attempt to optimize the empirical error by training in the search space.

This optimization problem can be solved by PSO, a stochastic global optimization method suitable for non-linear function optimization. We consider the entire ANN as a particle in a D -dimensional space. In other words, for instance, if an ANN has x inputs, y outputs and n_1 , n_2 , n_3 neurons in its three hidden layers respectively, then the number of dimensions (weights) for the particles is $d = x \times n_1 + n_1 \times n_2 + n_2 \times n_3 + n_3 \times y$.

Using this method, we create a population of particles that represent the weights of different ANNs. The other component of this optimization to be defined is the fitness function. A fitness function definition depends on the problem we aim to solve. In our experiment for classification, a fitness function is defined as feeding the entire training set to the ANN (one epoch) and adding up the number of correct classifications. To interpret the fitness values of particles, we assume that a particle with higher fitness value has less misclassification rates since the ANN weights have been represented as the dimensions of the particles.

In this variation of PSO the directions of flying particles are recorded at the beginning as positive or negative. When a particle reaches the maximum value of search space boundary, the particle position is reset to a random coordinate approximately in the middle of search space and its flying direction at this time will be changed to the opposite of that particle's original direction. In addition, if a particle's velocity reaches its maximum value and its coordinate is still within the search space, the particle's velocity is set to minimum value and its direction again is changed to the opposite direction. This updating method enables particles to explore the search space more thoroughly by experiencing broader ranges of possible values for both speed and location.

5. Results and discussion

We conducted four classification experiments using feed-forward ANNs with PSO based training. The two main parts of these experiments, PSO and the training of ANNs have been implemented in C++. The data sets were chosen from the UCI repository (Asuncion & Newman, 2007). The four data sets are Iris, Wine, SPECT heart and Ionosphere. The parameters values of ANNs and PSO used in the experiments are shown in Table 1. The information of attributes, training and validation data sets and target attributes (classes) is shown in Table 2. Since we are dealing with the two parameterized methods, PSO and ANNs, it is not feasible to set identical values for the parameters in every experiment because the type of problem that is solved greatly influences the values chosen for the parameters.

PSO parameters				
Dataset name	Iris	Wine	SPECT Heart	Ionosphere
Confidence factor (c_1, c_2)	1.7,1.7	1.7,1.7	1.7,1.7	1.7,1.7
Inertia(w)	0	0.9	0.9	0.9
maximum speed step (vmax)	1.0	1.2	1.2	1.2
Maximum value for each dimension (max-dim)	3.0	4.0	4.0	4.0
Iteration(T)	5000	1000	10000	1000
Particles(p)	20	30	20	30
MLP parameters				
Hidden layers	2	2	2	2
number of neurons in input, hidden and output layers respectively	4,40,28,1	13,15,10,1	22,30,5,1	34,15,10,1

Table 1. The parameters of PSO and Neural Networks used in the experiments

Dataset	Number of attributes /classes	Number of records in training/validation datasets	Number of correct classification	Accuracy MLP trained by PSO	Accuracy other methods
Iris	4/3	150/15	148	98.66%	85% - 97.77%
Wine	13/3	178/18	177	99.44%	96.1%-99.4%
SPECT Heart	22/2	80/187	172	91.97%	80% - 90.7%
Ionosphere	34/2	351/35	328	93.42%	90.7% - 96.7%

Table 2. The specifications of the data sets and the result of experiments

With the exception of the SPECT heart data set that contains a validation data set of 187 records, the rest of the data sets do not have any validation data sets. Therefore, we applied k -fold cross-validation method ($k=10$) for those data sets (Haykin, 1994). In k -fold cross-validation a data set is divided into k equal partitions where $k-1$ partitions are used as the training set and the remaining partition is used as a validation data set. The procedure is repeated k times with different partitions being used as the validation set each time, and the sum squared errors in k validations is considered as the final

error for the model. The results of 5 trainings from the total 10 trainings on the four data sets, Iris, Wine, SPECT heart and Ionosphere have been shown in the Tables 3-7. To avoid showing several graphs of the trainings, we have chosen only one graph of the ten trainings in each experiment, however, the related data is completely represented in the Tables 3-7.

In addition, to avoid recording unnecessary data, we ignored the fitness values that were not changed and were repeated in the intervals of iterations. Therefore, fitness values were recorded only when they were improved. The empty entries in the tables below show that after certain iterations the fitness had never improved until the maximum iteration number was reached. The data represented in Tables 3, 4, 6 can be useful to evaluate how fast and efficiently ANNs are trained. Moreover, by comparing the last fitness value of the trainings, shown in the mentioned tables, to their corresponding validation results shown in Table 5, we observe that a satisfactory result in the training does not provide a good result in the validation result and vice versa.

Training 5		Training 6		Training 7		Training 8		Training 9		Training 10	
	FIT	ITE	FIT	ITE	FIT	ITE	FIT	ITE	FIT	ITE	FIT
0	45	0	53	0	48	0	86	0	45	0	46
7	58	1	71	4	50	23	89	1	46	3	93
8	61	5	90	5	60	24	90	5	64	13	104
10	80	30	97	6	68	36	121	17	76	32	105
28	82	203	110	8	90	180	127	24	90	49	106
33	88	210	122	30	94	191	128	365	91	244	109
101	89	304	126	72	107	254	129	378	93	252	118
116	90	532	128	190	122	1707	130	452	94	466	121
368	91	540	129	210	124	4862	131	529	110	522	129
521	93	544	130	217	125	4963	132	754	120	853	132
604	104	573	131	584	126			872	127	922	133
745	116	1165	132	639	129			2012	128	1393	134
1066	126			735	130			3230	131		
2034	128			2374	131						
2035	129										
2043	130										
3343	131										

Table 3. The last 5 training results on Iris data set - ITE denotes the number of iterations and FIT stands for fitness.

One of the challenges in this experiment is that changing the parameter of PSO influences the performance of ANNs and vice versa. For instance, increasing the number of particles demands more fitness evaluation of particles and changing the number of hidden layers and their neurons adds more dimensions to the particles. In addition, it also slows down the performances. Therefore, there need to be some tradeoffs. The higher dimension we set, the more iterations are required to obtain a given accuracy. On the other hand, increasing the number of iterations adds more

computational load. Other difficulties arise when the number of samples and inputs of ANNs increase. In other word, increasing the number of records and attributes in a data set increases the training time.

Training 1		Training 2		Training 3		Training 4		Training 5	
ITE	FIT	ITE	FIT	ITE	FIT	ITE	FIT	ITE	FIT
0	108	0	108	0	76	0	92	0	96
13	114	2	115	2	93	1	133	3	99
50	140	8	121	5	107	229	144	48	100
65	141	14	136	20	111			90	102
96	145	66	143	24	114			91	104
168	146	143	149	29	117			98	111
210	148			44	130			103	114
311	149			51	140			105	115
334	150			74	150			107	138
500	151			233	151			193	140
550	152							450	147
802	153								
952	154								
966	155								

Table 4. The first 5 training results on Wine data set. - ITE denotes the number of iterations and FIT signifies the fitness

The other issue is the outputs chosen for the ANN. At the first glance, it might appear that the number of outputs should be equal to the number of classes. However, based on our experiment, we realized that this might not be the case necessarily. Therefore, we have chosen one output for ANN in all experiments and the activation function of the output neuron is set to a linear function.

Validations	1	2	3	4	5	6	7	8	9	10
Iris (Out of 15)	14	15	15	14	15	15	15	15	15	15
Wine (Out of 18)	18	18	18	18	17	18	18	18	18	18
Ionosphere (Out of 35)	35	31	30	31	35	34	31	35	35	35

Table 5. Validation results (number of correct classification) on Iris, Wine and Ionosphere data sets according to 5 trainings shown in tables 3,4,6.

Training 2		Training 3		Training 4		Training 5		Training 10	
ITE	FIT	ITE	FIT	ITE	FIT	ITE	FIT	ITE	FIT
0	244	0	234	0	225	0	225	0	219
2	254	4	240	1	245	1	234	7	232
5	263	8	241	4	248	2	244	10	242
70	264	9	250	17	251	6	248	13	244
72	266	25	251	39	258	8	254	19	256
86	270	27	260	46	274	49	270	45	257
87	273	50	273	72	275	335	272	52	262
115	274	53	274	75	278	371	275	63	264
124	279	68	276	79	279	415	277	65	265
158	280	78	277	83	281	556	278	71	266
180	283	83	278	700	287	662	279	85	269
181	284	84	280	802	290			119	270
280	286	202	281	990	291			122	271
311	287							183	272
318	288							211	274
								217	276
								224	278
								226	280
								286	281

Table 6. The training results of 2-5 and 10 on Ionosphere data set - ITE denotes the number of iterations and FIT signifies the fitness

By applying inertia parameter w in equation (1) the optimum point is reached faster. This is evident from the results of training shown in Tables 3 and 4 where the inertia parameter was not used for the trainings with Iris data set.

Iteration	0	1	3	8	21	28	37	133	218	254	556	1014	2659	3634
Fitness	40	42	47	50	54	57	59	61	63	64	65	66	67	68

Table 7. The result of training on 80 records of the SPECT heat dataset. The validation result is 172 records out of 187 which is equal to ,15 misclassification.

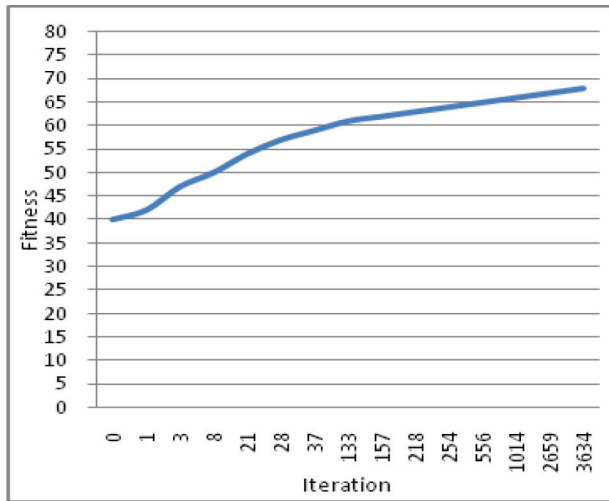


Fig. 3. Training on SPECT Heart

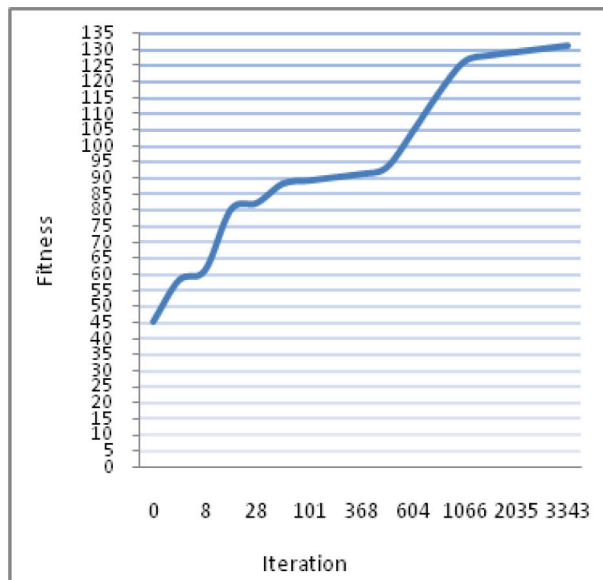


Fig. 4. Training on the 5th partition of Iris.

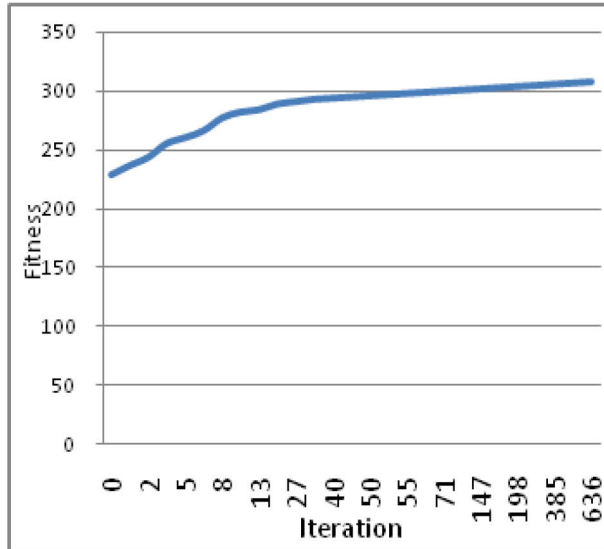


Fig. 5. Training on the 1st partition of Ionosphere

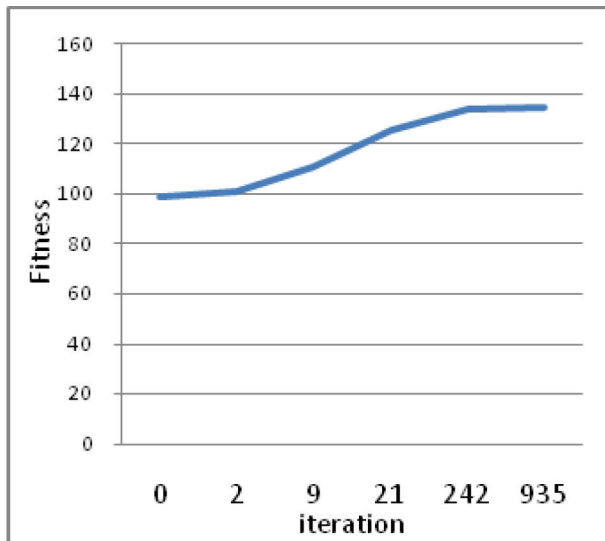


Fig. 6. Training on the 7th partition of Wine

6. Conclusion

PSO is a heuristic optimization method that performs well for different optimization problems. As with other optimization methods, it has not been proven that this method can always find the global optimum. However, it fulfills the exploitation and exploration of a search space. It is claimed that the local version of the method is able to find global

optimum. In the simulation, we experimented with training of feed-forward ANNs and demonstrated both the efficiency of this method and its competitiveness with other ANNs training methods. The proposed approach outperformed some of the previous results in our experiments. Moreover, the application of this method to train ANNs is limited by the structure of given ANNs. This means that with high numbers of inputs and neurons in the hidden layers, particle dimensions are increased and consequently training time rises. However, based on the properties of this method, it is possible to decrease the training time by parallel implementation. The feasibility of parallel implementation is a very important advantage of this method over other common training methods. Lastly, this method can be employed for training of ANNs with different topologies such as recurrent neural network where the gradient-based training methods are not suitable choices.

7. References

- Al-kazemi, B. & Mohan, CK (2002). Training feedforward neural networks using multi-phase particle swarm optimization, *Proceedings of the 9th International Conference on*, pp. 2615- 2619, 981-04-7524-1, USA, Nov. 2002, USA, IEEE, New York.
- Angelin P. J.,(1999).Using selection to improve particle swarm optimization, *Proceedings of IJCNN'99*, 0-7803-4869-9, pp. 84-89, USA, July 1999, IEEE,Anchorage.
- Asuncion A. & Newman D. J. (2007). UCI Machine Learning Repository, *Univ. of California, Irvine, School of Information and Computer Sciences*, 2007. [Online].available: <http://www.ics.uci.edu/~mllearn/MLRrepository.html>.
- Bergh, F. & Engelbrecht, A. (2002). A New Locally Convergent Particle Swarm Optimizer, *Proceedings of the IEEE International Conference on Conference on Systems, Man and Cybernetics*, pp. 96-101, 0-7803-7437-1, Oct. 2002, IEEE.
- Chunkai, Z., Yu, L. & Huihe, S. (2000). A new evolved artificial neural network and its application, *Proceedings of the 3rd World Congress on*, pp. 1065-1068, 0-7803-5995-X, June 2000, China, IEEE, Hefei.
- Eberhart, R. C. & Kennedy, J. (1995). A new optimizer using particle swarm theory, *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39-43, 0-7803-2676-8, Japan, Oct 1995, IEEE, Nagoya.
- Eberhart, R. C. & Shi, Y. (1998). Comparison between genetic algorithms and particle swarm optimization, *Proceedings of the 7th International Conference on Evolutionary Programming VII*, PP. 611-616, 3-540-64891-7, UK, 1998, Springer-Verlag, London.
- Fahlman, S. E. (1988). Faster-learning variations on back-propagation: an empirical study. *Proc. 1988 Connectionist Models Summer School*. D. S. Touretzky, G. E. Hinton and T. J. Sejnowski, (eds), Morgan Kaufmann, San Mateo, CA, 1988, pp. 38-51.
- Haykin S. (1994). *Neural networks, a comprehensive foundation*, Prentice Hall PTR, NJ, USA.
- Hecht-Nelson, R. (1989). Theory of the backpropagation neural network. *Proceedings of International Joint Conference on Neural Network*, vol. 1, pp. 593-605, Jun. 1989.
- Juang, C. F. (2004). A hybrid of genetic algorithm and particle swarm optimization for recurrent network design, *Systems, Man, and Cybernetics*, vol. 34, no. 2, pp. 997-1006, April 2004, IEEE.
- Kennedy, J. (2000). Stereotyping: Improving particle swarm optimization performance with cluster analysis, *Proceedings of the 2000 Congress on Evolutionary Computing*, pp. 1507-1512, USA, July 2000, IEEE, La Jolla.

- Mendes, R., Cortez, P., Rocha, M. & Neves, J.,(2002). Particle swarms for feedforward neural network training, *Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on*, pp. 1895-1899, 0-7803-7278-6, USA, May 2002, IEEE, Honolulu.
- Mitchell, M., *Introduction to Genetic Algorithms*, MIT Press, Cambridge, MA (1996).
- Riedmiller M. & Braun H., "A direct adaptive method for faster backpropagation learning: The rprop algorithm," in *Proc. IEEE Int. Conf. Neural Networks*, San Francisco, CA, Apr. 1993.
- Pu X., Z. Fang Z. & Liu Y. (2007). Multilayer perceptron networks training using particle swarm optimization with minimum velocity, *Proceedings of the 4th international symposium on Neural Networks: Advances in Neural Networks*, pp.237-245, 978-3-540-72394-3, China, 2007, Springer Berlin/Heidelberg, Nanjing.
- Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model, *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pp. 25-34, USA, July 1987, ACM, New York.
- Russ C., & Eberhart, Y. Shi. (2000). Comparing inertia weights and construction factors in particle swarm optimization, *Proceedings of the 2000 Congress on Evolutionary Computing*, pp. 84-88, USA, July 2000, IEEE, La Jolla.
- a) Shi, Y. & Eberhart, R. C. (1998). Parameter selection in particle swarm optimization, *Proceedings of the 7th International Conference on Evolutionary Programming VII*, pp. 591-600, 1998, UK, Springer-Verlag, London.
- b) Shi, Y. & Eberhart, R. C. (1998). A modified particle swarm optimizer, *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp.69-73, May 1998, USA, IEEE, Anchorage.
- Van den Bergh, F. & Engelbrecht, AP (2004). A Cooperative approach to particle swarm optimization, *Evolutionary Computation*, vol.8, No.3, (June 2004)(225-239), 1089-778X.

Resilient Back Propagation Algorithm for Breast Biopsy Classification Based on Artificial Neural Networks

Fawzi M. Al-Naima¹ and Ali H. Al-Timemy²

¹*Computer Engineering Department, College of Engineering,
Nahrain University, Baghdad,
Iraq,*

²*School of Computing, Communications and Electronics,
University of Plymouth, Plymouth, Devon,
United Kingdom*

1. Introduction

This chapter presents the classification of benign and malignant breast tumor based on Fine Needle Aspiration Cytology (FNAC) and Feed forward Neural Network (FFNN) trained with Resilient Back Propagation algorithm (RBP). Five hundred and sixty nine sets of cell nuclei characteristics obtained by applying image analysis techniques to microscopic slides of FNAC samples of breast biopsy have been used in this study. These data were obtained from the University of Wisconsin Hospitals, Madison. The dataset consist of thirty features which represent the input layer to the FFNN. The FFNN will classify the input features into benign and malignant. The sensitivity, specificity and accuracy were found to be equal 97.5%, 100% and 98.73% respectively. It can be concluded that RPB network gives fast and accurate classification and it works as promising tool for classification of breast cell nuclei.

Neural Networks (NN) derive their power due to their massively parallel structure, and an ability to learn from experience. They can be used for fairly accurate classification of input data into categories, provided they are previously trained to do so. The accuracy of the classification depends on the efficiency of training. The knowledge gained by the learning experience is stored in the form of connection weights, which are used to make decisions on fresh input (Al-Timemy et al., 2008).

One computer technique under investigation is the Artificial Neural Network (ANN) (Chester, 1993). Neural networks are tools for multivariate analysis that can be used to estimate disease risk. They are able to model complex nonlinear systems with significant variable interactions.

With one million new cases in the world each year, breast cancer is the commonest malignancy in women and comprises 18% of all female cancers. In the United Kingdom, where the age standardized incidence and mortality is the highest in the world, the incidence among women aged 50 approaches two per 1000 women per year, and the disease is the single commonest cause of death among women aged 40-50, accounting for about a

fifth of all deaths in this age group. There are more than 14000 deaths each year, and the incidence is increasing particularly among women aged 50-64, probably because of breast screening in this age group.

Of every 1000 women aged 50, two will recently have had breast cancer diagnosed and about 15 will have had a diagnosis made before the age of 50, giving a prevalence of breast cancer of nearly 2% (McPherson et al., 2000).

Conventional methods of monitoring and diagnosing the diseases rely on detecting the presence of particular features by a human observer. Due to large number of patients in intensive care units and the need for continuous observation of such conditions, several techniques for automated diagnostic systems have been developed in recent years to attempt to solve this problem. Such techniques work by transforming the mostly qualitative diagnostic criteria into a more objective quantitative feature classification problem (Ubeyli, 2007; Kordylewski et al., 2001; Kwak, & Choi, 2002; Ubeyli & Guler, 2005).

Breast cancer may be detected via a cautious study of clinical history, physical examination, and imaging with either mammography or ultrasound. However, definitive diagnosis of a breast mass can only be established through fine-needle aspiration (FNA) biopsy, core needle biopsy, or excisional biopsy (Chester, 1993). Among these methods, FNA is the easiest and fastest method of obtaining a breast biopsy, and is effective for women who have fluid-filled cysts. FNA uses a needle smaller than those used for blood tests to remove fluid, cells, and small fragments of tissue for examination under a microscope (Tingting & Nandi, 2007).

Research works on the Wisconsin diagnosis breast cancer (WDBC) data grew out of the desire of Dr. Wolberg to diagnose breast masses accurately based solely on FNA (Street et al. 1993; Wolberg et al., 1993; Wolberg et al., 1994). They applied image processing techniques to derive the WDBC dataset directly from digital scans of FNA slides (Wolberg et al., 1995). Then they employed machine learning techniques to differentiate benign from malignant samples (Mangasarian, 1995), which could be the earliest study of machine learning application to breast cancer detection. Several attempts have been proposed for diagnosis of the breast cancer that includes FFNN (Setiono, 2002), Radial Basis Function (RBF) network (Subashini et al., 2008), Self organization maps (Tingting & Nandi, 2007), fuzzy classifiers (Schaefer et al., 2008; Reyes et al., 1999; Mousa et al., 2005; Nauck, & Kruse, 1999; Abonyi & Szeifert, 2003), Linear Vector Quantization (LVQ) (Goodman et al., 2002), and Support Vector Mechanics (SVM) (Subashini et al., 2008; Akay, 2009; Polat & Gunes, 2007).

The purpose of this study is to develop a RBP network which will classify the breast biopsy samples into malignant and benign cysts based on the input features of cell nuclei characteristics. This network will act to help in the classification purposes of breast cancer.

2. The Classification Problem

The classification problem addressed in this study is the detection of malignant breast tumors from a set of benign and malignant samples, called the WDBC dataset, which was obtained from the University of Wisconsin Hospitals, Madison, available at <http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29>. Features in this dataset were computed from digitized FNA samples (Wolberg et al., 1993; Wolberg et al., 1994; Mangasarian, 1995), as follows:

After the FNA sample was taken from a breast mass, the material was mounted on a microscope slide and stained to highlight the cellular nuclei. A portion of well differentiated cells was scanned using a digital camera. The image analysis software system Xcyt was used to isolate individual nuclei. An approximate boundary of each nucleus was provided as input and taken to convergence to the exact nuclear boundary, using a semi-automatic segmentation procedure called "snakes". Beginning with a user defined approximate boundary as an initialization, the snake locates the actual boundary of the cell nucleus. In order to evaluate the size, shape and texture of each cell nuclei, the following ten characteristics were derived:

- (1) Radius is computed by averaging the length of radial line segments, which are lines from the center of mass of the boundary to each of the boundary points.
- (2) Perimeter is measured as the sum of the distances between consecutive boundary points.
- (3) Area is measured by counting the number of pixels on the interior of the boundary and adding one-half of the pixels on the perimeter, to correct for the error caused by digitization.
- (4) Compactness (*COM*) combines the perimeter and area to give a measure of the compactness of the cell, calculated as

$$COM = \frac{PER^2}{area} \quad (1)$$

This dimensionless number is minimized for a circle and increases with the irregularity of the boundary.

- (5) Smoothness (*SM*) is quantified by measuring the difference between the length of each radial line and the mean length of the two radial lines surrounding it. If this number is small relative to the distance between consecutive boundary points, then the contour is smooth in that region. To avoid the numerical instability associated with small divisors, the following equation is used to calculate the smoothness:

$$SM = \frac{\sum_{po\ int\ s} |r_i - (r_i + r_{i+1}) / 2|}{PER} \quad (2)$$

Where r_i is the length of the line from the center of mass of the boundary to each boundary point.

- (6) Concavity is captured by measuring the size of any indentations in the boundary of the cell nucleus.
- (7) Concave point is similar to concavity, but counts only the number of boundary points lying on the concave regions of the boundary, rather than the magnitude of such concavities.
- (8) Symmetry is measured by finding the relative difference in length between pairs of line segments perpendicular to the major axis of the contour of the cell nucleus. The major axis is determined by finding the longest chord, which passes from a boundary point through the center of the nucleus. The segment pairs are then drawn at regular intervals. To avoid numerically unstable results due to extremely small segments, the sums are again divided, rather than summing the quotients,

$$symmetry = \frac{\sum_i |left_i - right_i|}{\sum_i |left_i + right_i|} \quad (3)$$

where $left_i$ and $right_i$ denote the lengths of perpendicular segments on the left and on the right of the major axis, respectively.

(9) Fractal dimension is approximated using the “coastline approximation” described by Mandelbrot (Mandelbrot, 1997). The perimeter of the nucleus is measured using increasingly larger “rulers”. As the ruler size increases, the precision of the measurement decreases, and the observed perimeter decreases. Plotting these values on a log-log scale and measuring the downward slope gives the negative of an approximation to the fractal dimension.

(10) Texture is measured by finding the variance of the gray-scale intensities in the component pixels.

The mean value, standard error, and the extreme (largest or “worst”) value of each characteristic were computed for each image, which resulted in 30 features of 569 images, yielding a database of 569 X 30 samples representing 357 benign and 212 malignant cases.

3. Feed-Forward Neural Networks

A neural network is a parallel distributed information processing structure consisting of processing elements (neurons) interconnected via unidirectional signal channels called connections. Each processing element has a single output connection that branches into as many collateral connections as desired (Barbălată & Leuştean, 2004).

Neural networks develop information processing capabilities by learning for examples. Learning techniques can be roughly divided into two categories (Haykin, 1994); supervised learning and unsupervised learning.

Supervised learning requires a set of examples for which the desired network response is known. The learning process consists then in adapting the network in a way that it will produce the correct response for the set of examples. The resulting network should then be able to generalize (give a good response) when presented with cases not found in the set of examples.

In unsupervised learning the neural network is autonomous; it processes the data it is presented with, finds out about some of the properties of the data set and learns to reflect these properties in its output. What exactly these properties are, that network can learn to recognize, depends on the particular network model and learning method.

One of the most popular neural network paradigms is the feed-forward neural network. In a feed-forward neural network, the neurons are usually arranged in layers (Rumelhart et al., 1986). A feed-forward neural net is denoted as, $N_i \times N_1 \times \dots \times N_i \times \dots N_L \times N_o$

Where:

N_i represents the number of input units;

N_L represents the number of hidden layers;

N_i represents the number of units from the hidden layer I, $i=1,2,\dots,L$.

N_o represents the number of output units.

By convention, the input layer does not count, since the input units are not processing units, they simply pass on the input vector x . Units from the hidden layers and output layer are processing units. Figure 1 gives a typical fully connected 2-layer feed-forward network with a 3X4X3 structure.

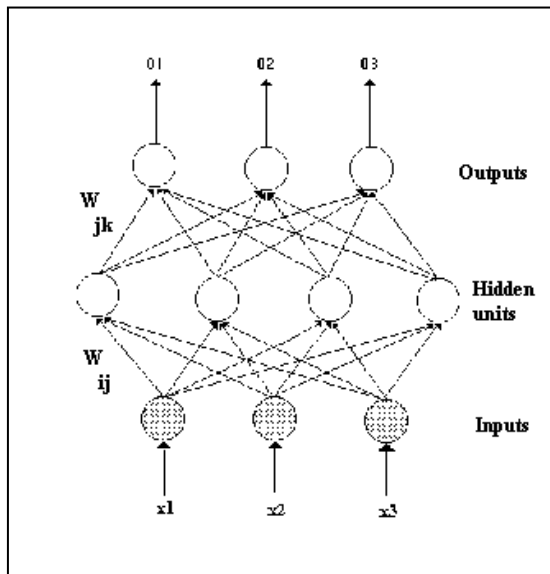


Fig. 1. A 3x4x3 feed-forward neural network.

Each processing unit has an activation function that is commonly chosen to be the sigmoid function:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (4)$$

The net input to a processing unit j is given by:

$$net_j = \sum_i w_{ij} x_i + \theta_j \quad (5)$$

Where x_i s are the outputs from the previous layer, w_{ij} is the weight (connection strength) of the link connecting unit i to unit j , and θ_j is the bias of unit j , which determines the location of the sigmoid function on the x -axis.

The activation value (output) of unit j is given by:

$$a_j = f(net_j) = \frac{1}{1 + e^{-net_j}} \quad (6)$$

The objective of different supervised learning algorithms is the iterative optimization of a so called error function representing a measure of the performance of the network. This error function is defined as the mean square sum of differences between the values of the output units of the network and the desired target values, calculated for the whole pattern set. The error for a pattern p is given by

$$E_p = \sum_{j=1}^{N_o} (d_{pj} - a_{pj})^2 \quad (7)$$

Where d_{pj} and a_{pj} are the target and the actual response value of output neuron j corresponding to the pattern p . The total error is,

$$E = \sum_{p=1}^P \frac{1}{2} E_p = \frac{1}{2} \sum_{p=1}^P \sum_{j=1}^{N_o} (d_{pj} - a_{pj})^2 \quad (8)$$

Where P is the number of the training patterns.

During the training process a set of pattern examples is used, each example consisting of a pair with the input and corresponding target output. The patterns are presented to the network sequentially, in an iterative manner, the appropriate weight corrections being performed during the process to adapt the network to the desired behavior. This iteration continues until the connection weight values allow the network to perform the required mapping. Each presentation of the whole pattern set is named an epoch.

One of the most popular supervised learning algorithms for feed-forward neural networks is Backpropagation (Rumelhart et al., 1986). In this algorithm the minimization of the error function is carried out using a gradient-descent technique. The necessary corrections to the weights of the network for each moment t are obtained by calculating the partial derivative of the network error function in relation to each weight w_{ij} . A gradient vector representing the steepest increasing direction in the weight space is thus obtained. The next step is to compute the resulting weight update. In its simplest form, the weight update is a scaled step in the opposite direction of the gradient. Hence, the weight update rule is :

$$\Delta_p w_{ij}(t) = -\varepsilon \cdot \frac{\partial E_p}{\partial w_{ij}}(t), \quad (9)$$

Where $\varepsilon \in (0,1)$ is a parameter determining the step size and is called the learning rate.

A momentum may be used with the idea of incorporating in the present weight update some influence of the past iteration. The weight update rule becomes:

$$\Delta_p w_{ij}(t) = -\varepsilon \cdot \frac{\partial E_p}{\partial w_{ij}}(t) + \alpha \cdot \Delta_p w_{ij}(t-1) \quad (10)$$

Where α is the momentum term which determines the amount of influence from the previous iteration to the present one.

4. The RBP Algorithm

The algorithm RBP introduced by M. Riedmiller in 1993 is a local adaptive learning scheme, performing supervised batch learning in feed-forward neural networks. The basic principle of RBP is to eliminate the harmful influence of the size of the partial derivative on the weight step. As a consequence, only the sign of the derivative is considered to indicate the direction of the weight update. To achieve this, we introduce for each weight w_{ij} its individual update-value $\Delta_{ij}(t)$, which solely determines the size of the weight-update (Riedmiller, 1993; Riedmiller & Braun, 1993).

It is introduced a second learning rule, which determines the evolution of the update-value $\Delta_{ij}(t)$. This estimation is based on the observed behavior of the partial derivative during two successive weight-steps:

$$\Delta_{ij}(t) = \begin{cases} \eta^+ \cdot \Delta_{ij}(t-1), & \text{if } \frac{\partial E}{\partial w_{ij}}(t) \cdot \frac{\partial E}{\partial w_{ij}}(t-1) > 0 \\ \eta^- \cdot \Delta_{ij}(t-1), & \text{if } \frac{\partial E}{\partial w_{ij}}(t) \cdot \frac{\partial E}{\partial w_{ij}}(t-1) < 0 \\ \Delta_{ij}(t-1), & \text{Otherwise} \end{cases} \quad (11)$$

Where

$$0 < \eta^- < 1 < \eta^+$$

In words, the adaptation rule works as follows. Every time the partial derivative of the corresponding weight w_{ij} changes its sign, which indicates that the last update was too big and the algorithm has jumped over a local minimum, the update-value $\Delta_{ij}(t)$ is decreased by the factor η^- . If the derivative retains its sign, the update-value is slightly increased in order to accelerate convergence in shallow regions.

Once the update-value for each weight is adapted, the weight-update itself follows a very simple rule: if the derivative is positive (increasing error), the weight is decreased by its update-value, if the derivative is negative, the update-value is added:

$$\Delta w_{ij}(t) = \begin{cases} -\Delta_{ij}(t), & \text{if } \frac{\partial E}{\partial w_{ij}}(t) > 0 \\ \Delta_{ij}(t), & \text{if } \frac{\partial E}{\partial w_{ij}}(t) < 0 \\ 0, & \text{else} \end{cases} \quad (12)$$

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t) \quad (13)$$

However, there is one exception. If the partial derivative changes sign that is the previous step was too large and the minimum was missed, the previous weight-update is reverted:

$$\Delta w_{ij}(t) = -\Delta w_{ij}(t-1), \quad (14)$$

$$\text{if } \frac{\partial E}{\partial w_{ij}}(t) \cdot \frac{\partial E}{\partial w_{ij}}(t-1) < 0$$

Due to that 'backtracking' weight-step, the derivative is supposed to change its sign once again in the following step. In order to avoid a double punishment of the update-value, there should be no adaptation of the update-value in the succeeding step. In practice this can be done by setting in the Δ_{ij} update-rule above:

$$\frac{\partial E}{\partial w_{ij}}(t-1) = 0$$

The partial derivative of the total error is given by

$$\frac{\partial E}{\partial w_{ij}}(t) = \frac{1}{2} \sum_{p=1}^P \frac{\partial E_p}{\partial w_{ij}}(t) \quad (15)$$

Hence, the partial derivatives of the errors must be accumulated for all P training patterns. This means that the weights are updated only after the presentation of all training patterns.

5. The Proposed FFNN Design

In the present work, the neural networks are used for the classification purposes. Three issues need to be settled in designing an ANN for a specific application:

- Topology of the network;
- Training algorithm;
- Neuron activation functions.

In our topology, the number of neurons in the input layer is 48 neurons for the ANN classifier. The output layer was determined by the number of classes desired. In our study, the output is either benign or malignant therefore; the output layer consists of one neuron. The hidden layer consists of twenty eight neurons. The general architecture of the proposed network is shown in Fig. 2.

Before the training process is started, all the weights are initialized to small random numbers. This ensured that the classifier network is not saturated by large values of the weights. In this experiment, the training set was formed by choosing 79 data sets for the testing process.

The tangent sigmoid (tansig) function was used as the neural activation function. The most important reason for choosing the sigmoid as an activation function for our networks is that the sigmoid function $f(x)$ is differentiable for all values of x , which allows the use of the powerful BP learning algorithm.

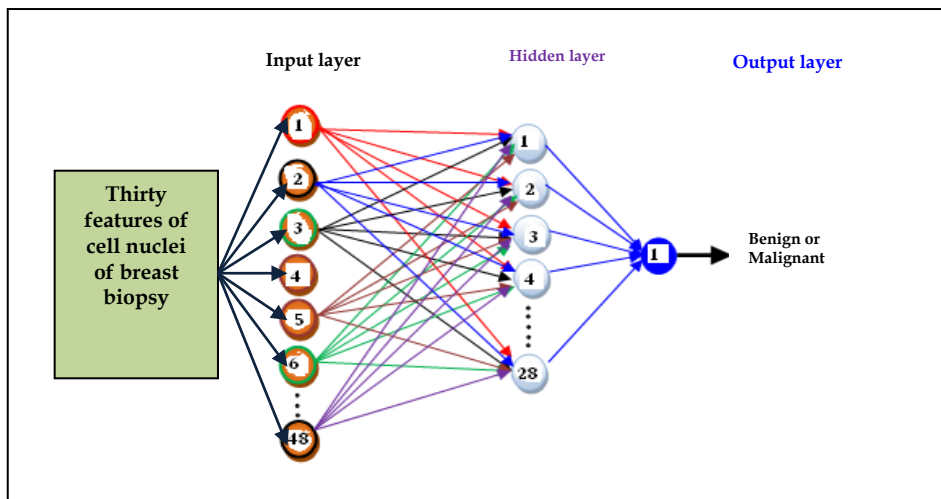


Fig. 2. General architecture of the proposed FFNN

The proposed network was trained with all 490 cases (317 benign and 173 malignant cases). These 490 cases are fed to the FFNN with 48 input neurons, one hidden layer of 28 neurons and one output neuron. MATLAB software package version 7 is used to implement the software in the current work. When the training process is completed for the training data (490 cases), the last weights of the network were saved to be ready for the testing procedure. Learning rate is set to 0.5, the output of the network was -1 for the class benign normal and 1 for the class malignant. The training algorithm used for this network is BPA. The performance goal was met at 2600 epochs after a training time of 67 sec.

The testing process is done for 79 cases (40 benign and 39 malignant). These 79 cases are fed to the proposed network and its their output is recorded for calculation of the sensitivity, specificity and accuracy of prediction.

The accuracy of the classification depends on the efficiency of training. The knowledge gained by the learning experience is stored in the form of connection weights, which are used to make decisions on fresh input.

6. Results and Discussion

The performance of the algorithm was evaluated by computing the percentages of Sensitivity (SE), Specificity (SP) and Accuracy (AC). The respective definitions of these parameters are as follows (Al-Timemy, 2008):

$$SE = \frac{TP}{(TP + FN)} \times 100 \quad (16)$$

$$SP = \frac{TN}{(TN + FP)} \times 100 \quad (17)$$

$$AC = \frac{(TP + TN)}{(TN + TP + FN + FP)} \times 100 \quad (18)$$

Where TP is the number of true positives, TN is the number of true negatives, FN is the number of false negatives, and FP is the number of false positives. Since it is interesting to estimate the performance of classifier based on the classification of benign and malignant breast cell nuclei, the true positives TP , false positives FP , true negatives TN , and false negatives FN are defined appropriately as shown below:

FP : Predicts benign as malignant.

TP : Predicts malignant as malignant.

FN : Predicts malignant as benign.

TN : Predicts benign as begin.

In our study, the output 1 indicates normal case. If the output is 2 this means that the patient may have abnormal kidney function. Sensitivity, specificity and accuracy of prediction have been calculated according to the above formals for all of the testing data (79 cases). Table 1 shows the resulted SE , SP and AC for testing data proposed networks.

	<i>No of cases</i>	<i>SE</i>	<i>SP</i>	<i>AC</i>
RBP	79	97.5%	100%	98.73%

Table 1. The results after training of the proposed network

From the table, the obtained accuracy means that there was only one misidentification. This is regarded a very good and the system is reliable. The results showed that the algorithm can be reliable purposes in the classification purposes.

In practice, the number of neurons in the hidden layer varies according to the specific recognition task and is determined by the complexity and amount of training data available. If too many neurons are used in the hidden layer, the network will tend to memorize the data instead of discovering the features. This will result in failing to classify new input data. The goal error was 0.01 and the network reached this error after 2600 epochs. Fig.3 displays the error of the training of the network versus epoch's number.

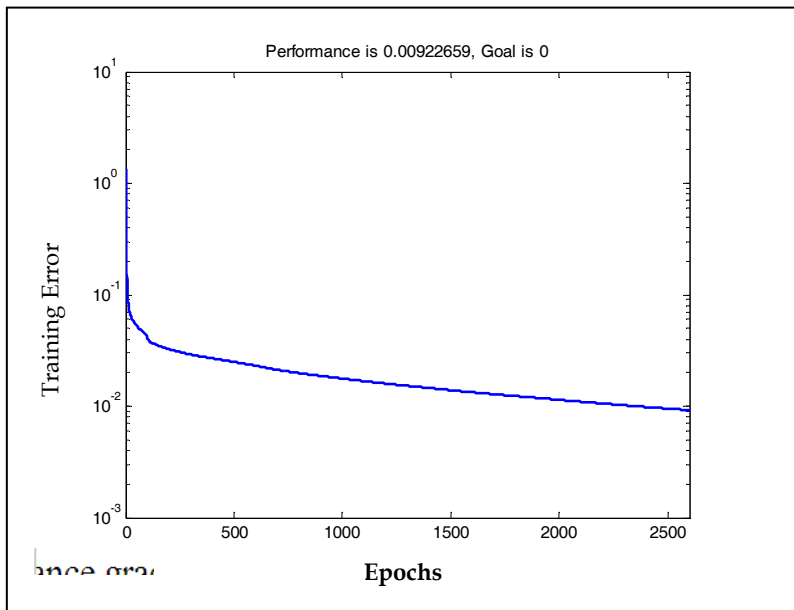


Fig. 3. Error of the training versus epochs

7. Conclusion

In this chapter, Resilient BPA has been implemented for classification of benign from malignant breast tumor. Five hundred and sixty nine sets of cell nuclei characteristics obtained by applying image analysis techniques to microscopic slides of FNAC samples of breast biopsy have been used in the current work. MATLAB software package version 7 is used to implement the software in the current work. These feature vectors which consist of thirty image analysis features each were carried out to generate training and testing of the proposed Neural Network. The accuracy is calculated to evaluate its effectiveness of the proposed network. The obtained accuracy of the network was 98.73% whereas the sensitivity and specificity were found to be equal 100% and 98.73% respectively. It can be concluded that the proposed system gives fast and accurate classification of breast tumors.

Acknowledgment

The authors would like to express their thanks to Dr. William H. Wolberg, W. Nick Street and Olvi L. Mangasarian for providing Breast Cancer Wisconsin (Diagnostic) Data Set used in this work. They are with the University of Wisconsin, 1210 West Dayton St., Madison, WI 53706 USA.

8. References

- Abonyi, J. & Szeifert, F. (2003). Supervised fuzzy clustering for the identification of fuzzy classifiers. *Pattern Recognition Letters*, Vol. 14, No. 24, 2003, pp. 2195–2207.
- Akay, M. F. (2009). Support vector machines combined with feature selection for breast cancer diagnosis, *Expert Systems with Applications*, Vol. 36, 2009, pp 3240–3247.
- Al-Timemy, A. H. A. (2008). Self-organization maps for prediction of kidney dysfunction, *Proc. 16th Telecommunications Forum TELFOR*, Belgrade, Serbia, 2008.
- Al-Timemy, A. H. A., Al-Naima, F. M and Mahdi, S. (2008). Data acquisition system for myocardial infarction classification based on wavelets and neural networks," *Proc. of the Fifth International Multi-Conference on Systems, Signals and Devices (IEEE SSD'08)*, Amman, Jordan.
- Barbălată, C. & Leuştean, L. (2004). Average monthly liquid flow forecasting using neural networks, *Draft paper*, 2004.
- Chester, M. (1993). *Neural Networks: A Tutorial*, Englewood Cliffs, NJ: Prentice and Hall, ch.2.
- Goodman, D. E., Boggess, L. & Watkins, A. (2002). Artificial immune system classification of multiple-class problems, *Proceedings of the Artificial Neural Networks in Engineering* (pp. 179–183).
- Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation*, New York: Macmillan.
- Kordylewski, H., Graupe D. & Liu, K. (2001). A novel large-memory neural network as an aid in medical diagnosis applications, *IEEE Transactions on Information Technology in Biomedicine*, Vol. 5, No. 3, 2001, pp. 202–209.
- Kwak, N. & Choi, C. H. (2002). Input feature selection for classification problems, *IEEE Transactions on Neural Networks*, Vol. 13, No. 1, 2002, pp. 143–159.
- Mandelbrot, B. B. (1997). *The Fractal Geometry of Nature*, W. H. Freeman and Company, New York.
- Mangasarian, O. L. (1995). Breast cancer diagnosis and prognosis via linear programming, *Cancer Lett.*, Vol. 43, No. 4, 1995, pp. 570–577.
- McPherson, K., Steel, C. M. & Dixon, J. M. (2000). ABC of Breast Diseases, Breast cancer – epidemiology, risk factors, and genetics, *BMJ*, Vol. 321, No. 9, September 2000, pp. 624–628.
- Mousa, R, Munib, Q. & Moussa, A. (2005). Breast cancer diagnosis system based on wavelet analysis and fuzzy-neural", *Expert Systems with Applications*, Vol. 28 ,2005, pp. 713–723.
- Nauck, D. & Kruse, R. (1999). Obtaining interpretable fuzzy classification rules from medical data", *Artificial Intelligence in Medicine*, Vol. 16, 1999, pp. 149–169.
- Polat K. & Gunes, S. (2007). Breast cancer diagnosis using least square support vector machine, *Digital Signal Processing*, Vol. 17, No. 4, 2007, pp. 694–701.
- Reyes, P., Andre's, C. & Sipper, M. (1999). A fuzzy-genetic approach to breast cancer diagnosis, *Artificial Intelligence in Medicine*, Vol. 17, 1999, pp.131–155.
- Riedmiller, M (1993). Untersuchungen zu Konvergenz und Generalisierungs-verhalten überwachter Lernverfahren mit dem SNNS, in A. Zell, editor, *SNNS 1999 Workshop Proceedings*, Stuttgart, September 1993.
- Riedmiller, M. & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: The RPROP algorithm, in H. Ruspini, editor, *Proceedings of the IEEE International Conference on Neural Networks (ICNN)*, San Francisco, USA, pp. 586–591.

- Rumelhart, E., Hinton, G. E. & Williams, R. J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*; Vol. 1: Foundations. The MIT Press, Cambridge, Massachusetts.
- Schaefera, G., Závisekb, M. & Nakashima, T. (2008). Thermography based breast cancer analysis using statistical features and fuzzy classification, *Pattern Recognition*, doi:10.1016/j.patcog.2008.08.007.
- Setiono, R. (2002). Generating concise and accurate classification rules for breast cancer diagnosis, *Artificial Intelligence in Medicine*, Vol. 18, No. 3, 2002, pp. 205–217.
- Street, W. N., Wolberg, W. H. & Mangasarian, O. L. (1993). Nuclear feature extraction for breast tumor diagnosis, *In proceedings of IST/SPIE Symposium on Electronic Imaging: Science and Technology*, vol. 1905, San Jose, CA, 1993, pp. 861–870.
- Subashini, T. S., Ramalingam, V. & Palanivel, S. (2008). Breast mass classification based on cytological patterns using RBFNN and SVM, *Expert Systems with Applications*, doi:10.1016/j.eswa.2008.06.127.
- Tingting, Mu. & Nandi, A. K. (2007). Breast cancer detection from FNA using SVM with different parameter tuning systems and SOM-RBF classifier, *Journal of the Franklin Institute*, Vol. 344, 2007, pp. 285–311.
- Ubeyli, E. D. (2007). Implementing automated diagnostic systems for breast cancer detection, *Expert Systems with Applications*, Vol. 33, 2007, pp. 1054–1062.
- Ubeyli, E. D. & Guler, I. (2005). Feature extraction from Doppler ultrasound signals for automated diagnostic systems. *Computers in Biology and Medicine*, Vol., No. 9 2005, pp. 735–764.
- Wolberg, W. H., Street, W. N. & Mangasarian, O. L. (1993). Breast cytology diagnosis via digital image analysis, *Anal. Quant. Cytol. Histol.*, Vol 15, No. 6 (1993), pp. 396–404.
- Wolberg, W. H., Street, W. N. & Mangasarian, O. L. (1994). Machine learning techniques to diagnose breast cancer from fine-needle aspirates, *Cancer Lett.*, Vol. 77 1994, pp. 163–171.
- Wolberg, W. H., Street, W. N., & Mangasarian, O. L. (1995). Image analysis and machine learning applied to breast cancer diagnosis and prognosis, *Anal. Quant. Cytol. Histol.*, Vol. 17, No.2 1995, pp. 77–87.

SIMD Architecture Approach to Artificial Neural Networks Realisation

Jacek Mazurkiewicz

*Institute of Computer Engineering, Control and Robotics
Wroclaw University of Technology
ul. Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, POLAND*

1. Introduction

A new methodology for artificial neural networks implementation based on SIMD architecture is described. The method proposed is based on pipelined systolic arrays. The partial parallel realisation of learning and retrieving algorithms using unified processing structure is assumed. The discussion is realized based on operations which create the following steps of weight tuning and answer generation algorithms. The data which are transferred among the calculation units are the second criterion of the problem. The results of discussion show that it is possible to create the universal structure to implement all algorithms related to Hopfield, Kohonen and Hamming neural networks. Theoretical foundations for synthesis of digital devices related to neural network algorithms are the main goal. The number of neurons is unlimited, but necessary calculations can be done using reduced number of elementary processors. This way the dependability features of the proposed methodology are focused to Fault Tolerant Computing approach. The evaluation of efficiency measures related to the proposed structures is the final step of described work. The elaboration can be divided into following stages:

- accommodation of analysed algorithms to partial parallel realisation in systolic array;
- synthesis of constructional unified systolic structures and data flow description for analysed algorithms;
- proposals of functions for processing elements;
- evaluation of proposed solutions.

Proposed systolic arrays are based on the set of assumptions:

- results obtained as the output of structures are convergent to theoretical description of proper algorithms of artificial neural nets;
- proposed structures are realised based on digital elements only;
- number of neurons limits maximum number of elementary processors used in structure.

The proposed solutions are characterised by very universal approach with partial parallel information processing which reduces calculation time in significant way. The work can be used as base point both for dedicated circuits to implement artificial neural networks and for structures which realise the net by ready-to-use elements with point-to-point communication rules as PLD for example.

2. Hopfield neural network algorithms

The binary Hopfield net has a single layer of processing elements, which are fully interconnected - each neuron is connected to every other unit. Each interconnection has an associated weight: w_{ji} is the weight to unit j from unit i . In Hopfield network, the weight w_{ij} and w_{ji} has the same value. Mathematical analysis has shown that when this equality is true, the network is able to converge. The inputs are assumed to take only two values: 1 and 0. The network has N nodes containing hard limiting nonlinearities. The output of node i is fed back to node j via connection weight w_{ij} .

2.1 Retrieving phase

During the retrieving algorithm each neuron performs the following two steps (Mazurkiewicz, 2003b) (Ferrari & Ng, 1992):

- computes the coproduct:

$$\varphi_p(k+1) = \sum_{j=1}^N w_{pj} v_j(k) - \theta_p \quad (1)$$

w_{pj} - weight related to feedback signal, $v_i(k)$ - feedback signal, θ_p - bias
- updates the state:

$$v_p(k+1) = \begin{cases} 1 & \text{for } \varphi_p(k+1) > 0 \\ v_p(k) & \text{for } \varphi_p(k+1) = 0 \\ -1 & \text{for } \varphi_p(k+1) < 0 \end{cases} \quad (2)$$

The process is repeated for the next iteration until convergence, which occurs when none of the elements changes state during any iteration. The initial and the end conditions for the iteration procedure require the following equations: (Mazurkiewicz, 2004)

$$\forall_p v_p(k+1) = v_p(k) = y_p \quad \forall_p v_p(0) = x_p \quad (3)$$

2.2 Hebbian learning algorithm

The training patterns are presented one by one in a fixed time interval. During this interval, each input data is communicated to its neighbour N times:

$$w_{ij} = \begin{cases} \frac{1}{N} \sum_{m=1}^M x_i^{(m)} x_j^{(m)} & \text{for } i \neq j \\ 0 & \text{for } i = j \end{cases} \quad (4)$$

M - number of training vectors

2.3 Delta-rule learning algorithm

The weights are calculated in recurrent way including all training patterns, according to the following matrix equation:

$$W = W + \frac{\eta}{N} [x^{(i)} - Wx^{(i)}] [x^{(i)}]^T \quad (5)$$

$\eta \in [0,7, 0,9]$ - learning rate, N - number of neurons, W - matrix of weights, x - input vector
The learning process stops when the next training step generates the changes of weights which are less then the established tolerance ε .

3. Systolic arrays for Hopfield neural network

Systolic arrays are prepared based on proper Data Dependence Graphs - directed graphs that specify the data dependencies of algorithms. In a Data Dependence Graph nodes represent computations and arcs specify the data dependencies between computations. (Ferrari & Ng, 1992) (Mazurkiewicz, 2004)

3.1 Idea of systolic array for Hebbian learning algorithm

Each node in Data Dependence Graph for Hebbian training algorithm multiplies two of corresponding input signals x_i and obtains this way the weight w_{ij} which is stored in local memory unit. So it realizes three operations. Each elementary processor is responsible for these three operations. The input signals x_i are passed to the nearest bottom neighbours and the neighbours on the right hand (Fig.1).

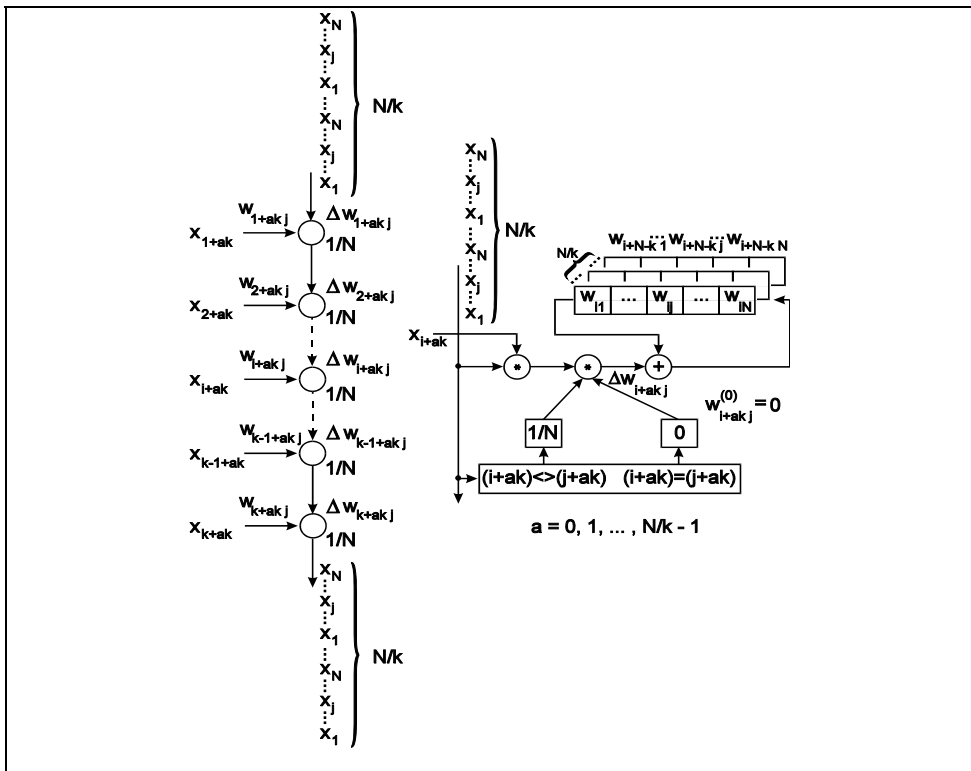


Fig. 1. Idea of systolic array for Hebbian learning algorithm

3.2 Idea of systolic array for Delta-rule learning algorithm

The Data Dependence Graph for Delta-rule training algorithm we can divide into two parts: *Relation Graph* G_r and *Value Graph* G_w . Each node which belongs to G_r multiplies a corresponding input signal x_i and weight value w_{ij} , then it subtracts the multiplication result from the input signal x_i . Each node in the G_w part of the Data Dependence Graph is responsible for three operations. During the first operation the node multiplies the corresponding result obtained at the end of the calculations related to the G_r part of the Data Dependence Graph and the input signal x_i . During the second operation each node multiplies the obtained values and the fraction: learning rate/number of neurons. At the end the values of weights are upgraded. This way the weights are obtained and next they are stored in local memory unit. The input signals x_i are passed to the nearest bottom neighbours (Kung, 1993). Operations described by *Relation Graph* G_r and *Value Graph* G_w ought to be realized in sequence – so processor executes five basic operations. (Fig. 2).

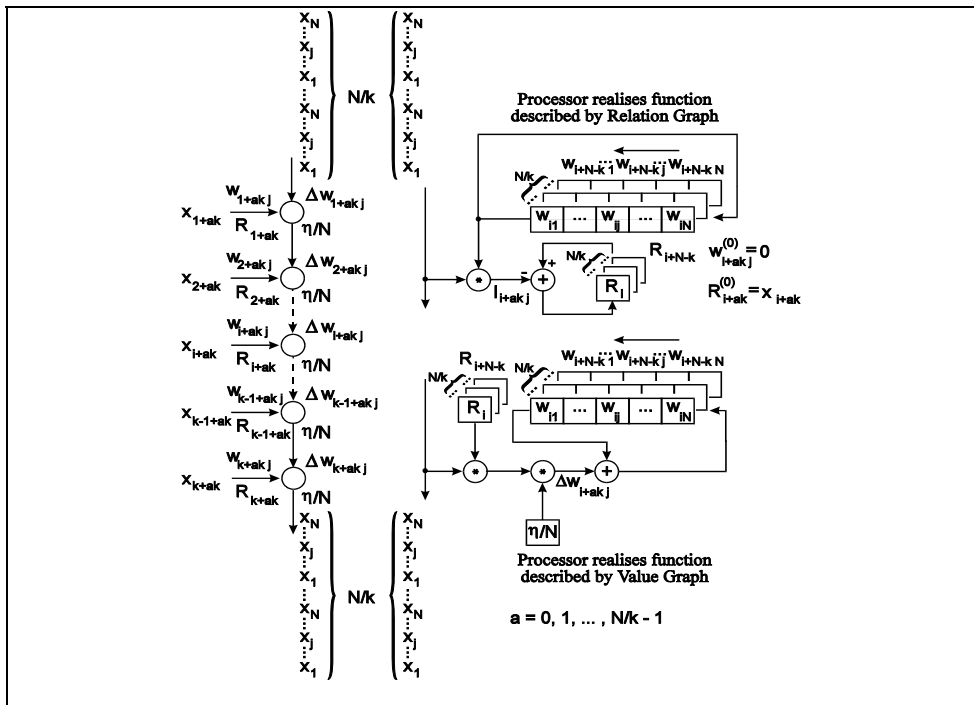


Fig. 2. Idea of systolic array for Delta-rule learning algorithm

3.3 Idea of systolic array for retrieving algorithm

Each node in Data Dependence Graph for retrieving algorithm multiplies the input signal x_i or feedback signals v_i and corresponding weight w_{ij} which is stored in local memory unit. The product of multiplication is passed to the nearest neighbour on the right hand (two basic operations). The ϕ_i nodes collect the partial products and calculate the global value of coproduct. The last nodes on the right are the comparators to check if the next iteration is necessary. (Fig. 3).

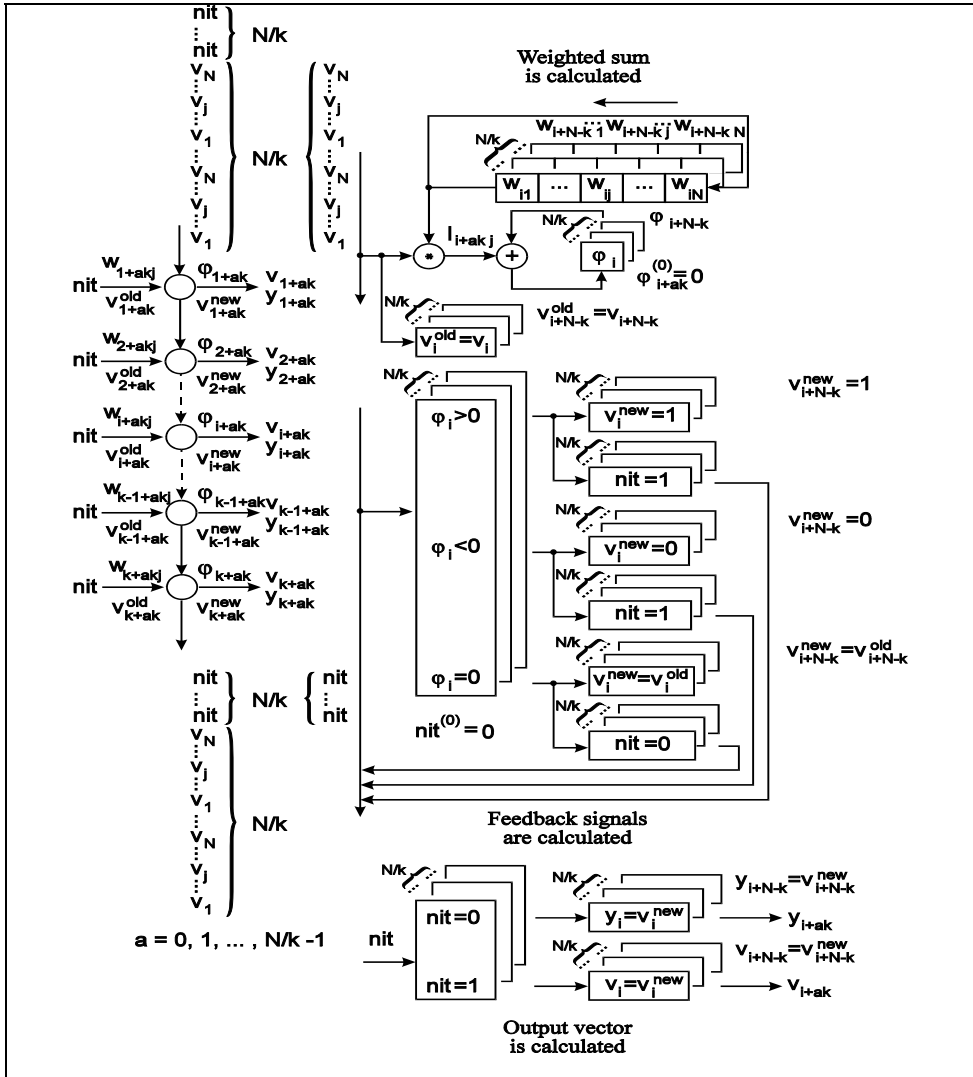


Fig. 3. Idea of systolic array for retrieving algorithm

4. Efficiency of systolic arrays for Hopfield neural network

4.1 Computation time and Block period

This is time between starting the first computation and finishing the last computation of problem. Given a coprime schedule vector \vec{s} , the computation time equals (Kung, 1993), (Zhang, 1999):

$$T = \max_{\vec{p}, \vec{q} \in L} \{ \vec{s}^T (\vec{p} - \vec{q}) \} + 1 \tag{6}$$

L - is the index set of the nodes in the Data Dependence Graph

Block Period is time interval between the initiation of two successive blocks of operations (Kung, 1993), (Zhang, 1999). In the presented architectures for all algorithms the schedule vector is defined as: $\vec{s} = [1, 1]$. For Hebbian training implementation - taking number of basic operations into account - we can calculate the computation time and block period as:

$$T_{\text{ystol}} = 3N \left(\frac{N-1}{K} + 1 \right) \tau M \quad T_{\text{block}} = 3N \left(\frac{N-1}{K} + 1 \right) \tau \quad (7)$$

In fact the Data Dependence Graph for this algorithm is combined by two independent structures of operations. The computation time for both parts of algorithm isn't the same because the number of basic operations is different, number of nodes and the topology of them is the same. Additionally the estimated computation time ought to be modified by number of iterations related to single training pattern:

$$T_{\text{ystol}} = 5N \left(\frac{N-1}{K} + 1 \right) \tau M \beta \quad T_{\text{block}} = 5N \left(\frac{N-1}{K} + 1 \right) \tau \beta \quad (8)$$

τ - processing time for elementary processor, M - number of training patterns, β - number of iterations for single training pattern, K - number of elementary processors.

The retrieving algorithm also requires multiple presentation of each pattern but single retrieving procedure doesn't require all patterns at the same time. The computation time we can describe using the following equations:

$$T_{\text{ystol}} = 2N \left(\frac{N+1}{K} + 1 \right) \tau \beta \quad T_{\text{block}} = 2N \left(\frac{N+1}{K} + 1 \right) \tau \quad (9)$$

In all equations we can find parameter denoted as K - the number of elementary processors. This way the FTC parameter of the structure can be discussed. The definition of SIMD architecture guarantees the unique construction and function of processors - so if we can observe the changes of efficiency parameters related to the number of used processors we can say a lot of the results of the failures. But we have to remember that using only single processor it is possible to realize the whole calculation process. Of course the efficiency parameters will be very poor, but the structure is still working.

4.2 Pipelining period

This is the time interval between two successive computations in a processor. As previously discussed, if both \vec{d} and \vec{s} are irreducible, then the pipelining period equals to:

$$\alpha = \vec{s}^T \vec{d} \quad (10)$$

The pipelining period is the same for all algorithms - equals to: $\alpha = 1$ - is as short as possible (Kung, 1993) (Ferrari & Ng, 1992) (Shiva, 1996).

4.3 Speed-up and Utilization rate

Lets define the speed-up factor as the ratio between the sequential computation time T_{seq} and the array computation time T_{systol} and the utilization rate as the ratio between the speed-up factor and the number of processors (Kung, 1993).

$$speed - up = \frac{T_{seq}}{T_{systol}} \quad utilization\ rate = \frac{speed - up}{K} \quad (11)$$

Sequential computation time for Hopfield neural network algorithms - taking number of neurons, number of weights and number of basic operations into account - equals:

- for Hebbian learning:

$$T_{seq} = 3N^2 \tau M \quad (12)$$

- for Delta-rule learning:

$$T_{seq} = 5N^2 \tau M \beta \quad (13)$$

- for retrieving algorithm:

$$T_{seq} = 2N(N + 2) \tau \beta \quad (14)$$

Based on values of array computation time calculated in chapter 4.1. we can evaluate:

- for Hebbian learning:

$$speed - up = \frac{NK}{N-1+K} \quad utilization\ rate = \frac{N}{N-1+K} \quad (15)$$

- for Delta-rule learning:

$$speed - up = \frac{NK}{N-1+K} \quad utilization\ rate = \frac{N}{N-1+K} \quad (16)$$

- for retrieving algorithm:

$$speed - up = \frac{(N+2)K}{N+1+K} \quad utilization\ rate = \frac{N+2}{N+1+K} \quad (17)$$

The classical parameters calculated for presented architecture are also related to the number of active processors. We can model the speed-up and utilization rate in function of not-failed processors - we can control the efficiency in case of FTC features of proposed architecture.

5. Kohonen neural network algorithms

5.1 Learning algorithm

The learning algorithm is based on the Grossberg rule (Mazurkiewicz, 2005a) (Mazurkiewicz, 2005b). All weights are modified according to the following equation:

$$w_{ij}(k+1) = w_{ij}(k) + \eta(k)\Lambda(i^w, j^w, i, j)(x_l - w_{ij}(k)) \quad (18)$$

k - iteration index, η - learning rate function, x_l - component of input learning vector, w_{ij} - weight associated with connection from component of input learning vector x_l and neuron indexed by (i, j) , Λ - neighborhood function, (i^w, j^w) - indexes related to winner neuron, (i, j) - indexes related to single neuron from Kohonen map.

The learning rate η we assume as a linear decreasing function. Learning rate function is responsible for the number of iterations - it marks the end of learning process. The presented solution is based on the following description of the neighborhood function (Asari & Eswaran, 1992):

$$\Lambda(i^w, j^w, i, j) = \begin{cases} 1 & \text{for } r = 0 \\ \frac{\sin(ar)}{ar} & \text{for } r \in \left(0, \frac{2\pi}{a}\right) \\ 0 & \text{for other values } r \end{cases} \quad (19)$$

a - neighborhood parameter, r - distance from winner neuron to each single neuron from Kohonen map, calculated by indexes of neurons as follow:

$$r = \sqrt{(i^w - i)^2 + (j^w - j)^2} \quad (20)$$

The learning procedure is iterative: weights are initialized by random values; position of winner neuron for each learning vector is calculated by ordinary Kohonen retrieving algorithm using random values of weights; weights are modified using Grossberg rule (18); the learning rate is modified, the neighborhood parameter a (19) is modified and if the learning rate is greater than zero weights are modified by the next learning vector, else the learning algorithm stops (Mazurkiewicz, 2003a).

5.2 Retrieving algorithm

During the retrieving phase the Euclidean distance: the weights vector and the output vector is calculated. The winner neuron is characterized by the shortest distance (Mazurkiewicz, 2005b) (Mazurkiewicz, 2003a). Each neuron from Kohonen map calculates the output value according to the classical weighted sum:

$$Out(i, j) = \sum_{l=0}^{N-1} x_l w_{ij} \quad (21)$$

$Out(i, j)$ - output value calculated by single neuron of Kohonen map indexed by (i, j)

6. Data Dependence Graphs for Kohonen neural network

6.1 Kohonen learning algorithm

For 1-D Kohonen map neurons are placed in single line, each neuron has two neighbors, excluding neurons at the ends of line. For such topology there are $(N \times K)$ weights if we assumed N -element input vector and K neurons which create the Kohonen map. 1-D Kohonen map ought to be described by rectangular Data Dependence Graph (Fig. 6.). Each node of the graph is responsible for single weight calculation. using Grossberg rule (18)

(Fig. 4). The current value of the weight is stored in the local memory of each node. The node decreases the learning rate in automatic way. The size of the graph equals to the size of the weight matrix. Each node of the graph is loaded by two signals. The neighborhood function is calculated using sinus function. We propose to place the values of sinus in a table and store them in a local memory of each node. The neighborhood parameter a (19) is also stored in the local memory and is sequential reduced by negative counter.

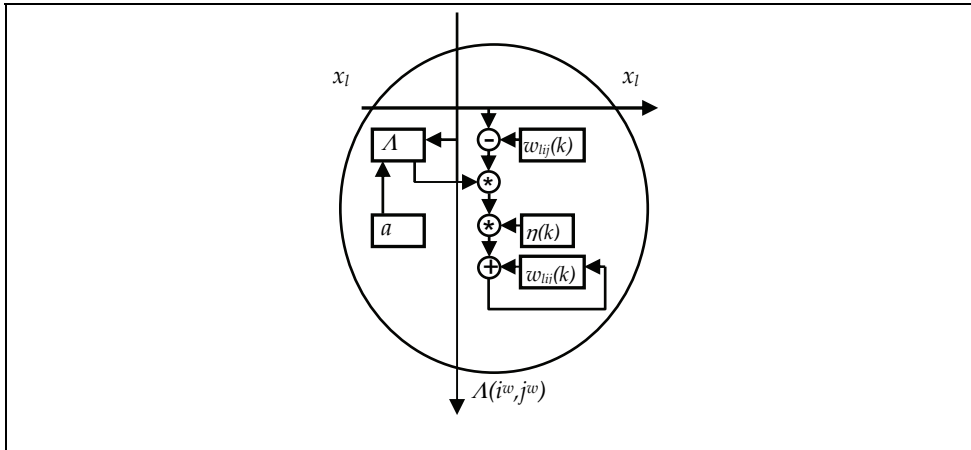


Fig. 4. Single node of Data Dependence Graph for learning algorithm

6.2 Kohonen retrieving algorithm

1-D Kohonen map is described by rectangular Data Dependence Graph (Fig. 7.). Each node of the graph calculates the component of the weighted sum (21) (Fig. 5.). The necessary weight value is stored in a local memory of the node. The size of the graph equals to the size of the weight matrix.

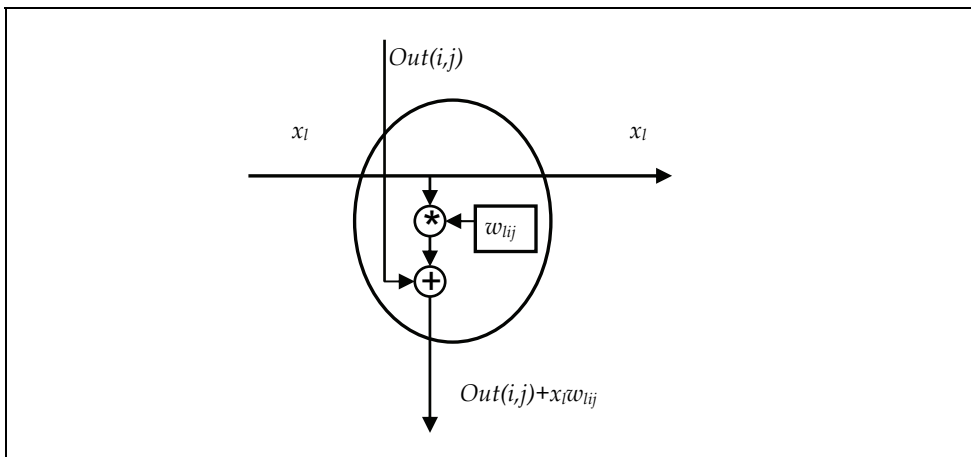


Fig. 5. Single node of Data Dependence Graph for retrieving algorithm

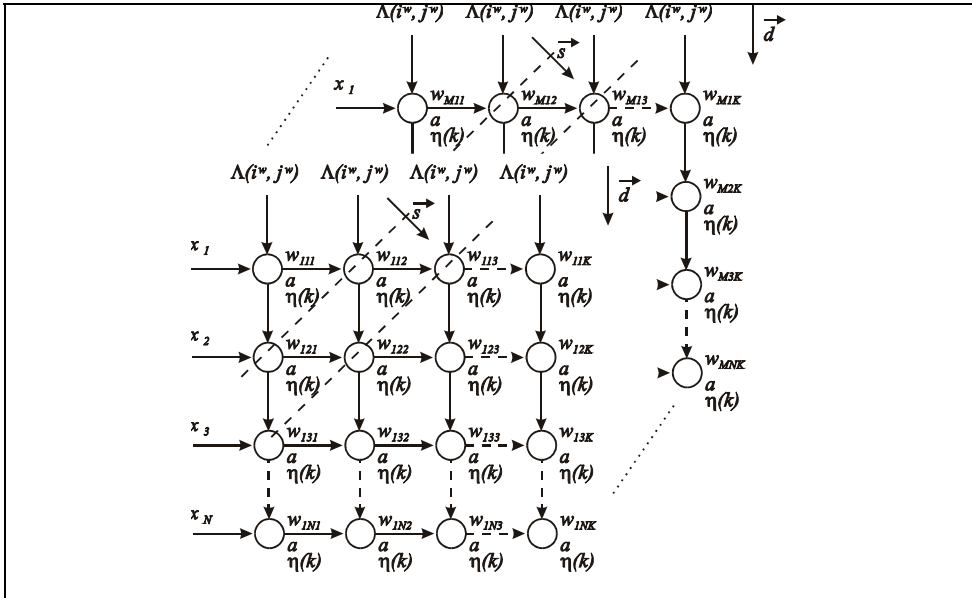


Fig. 6. Data Dependence Graph for learning algorithm of Kohonen map

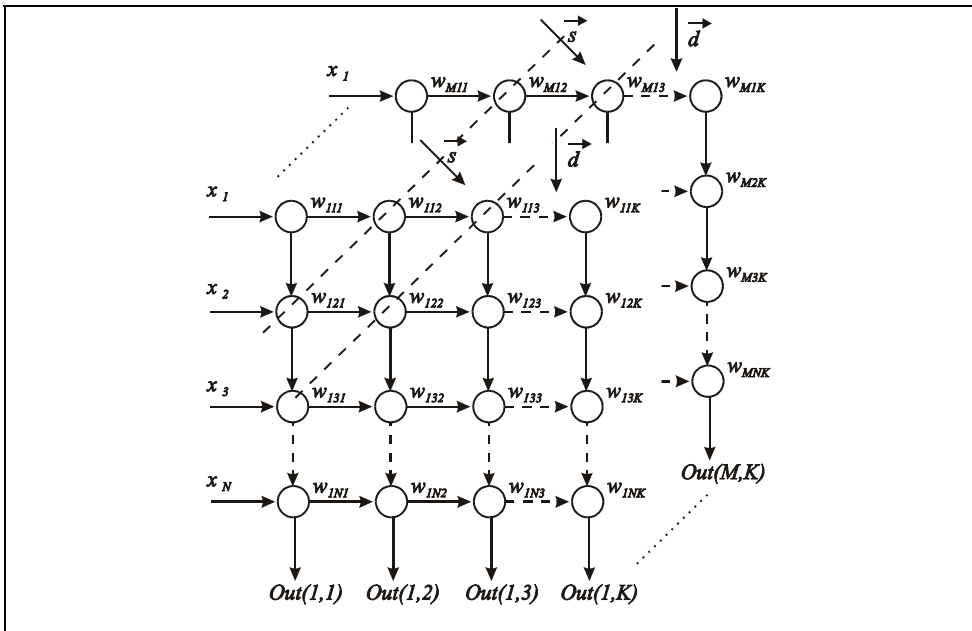


Fig. 7. Data Dependence Graph for retrieving algorithm of Kohonen map

7. Mapping Data Dependence Graphs onto systolic array

The Data Dependence Graphs for retrieving and learning algorithms are local and composed by the same number of nodes. The single neuron operations are described by the column of the graph (Mazurkiewicz, 2003a). Multi-dimensional Kohonen map is described by the set of 1-D Data Dependence Graphs (Fig. 6.) (Fig. 7.). It means that the slabs work in parallel (Asari & Eswaran, 1992) (Mazurkiewicz, 2003a) (Mazurkiewicz, 2003b). The graphs can be converted to an universal structure able to implement learning algorithm as well as retrieving algorithm using processors with switched functions (Fig. 8.) (Kung, 1993) (Asari & Eswaran, 1992). The systolic arrays are the result of the linear projection of Data Dependence Graphs onto lattice of points, known as processor space. The elementary processor combines operations described by nodes taken from single vertical line of the graph (Zhang, 1999) (Petkov, 1993).

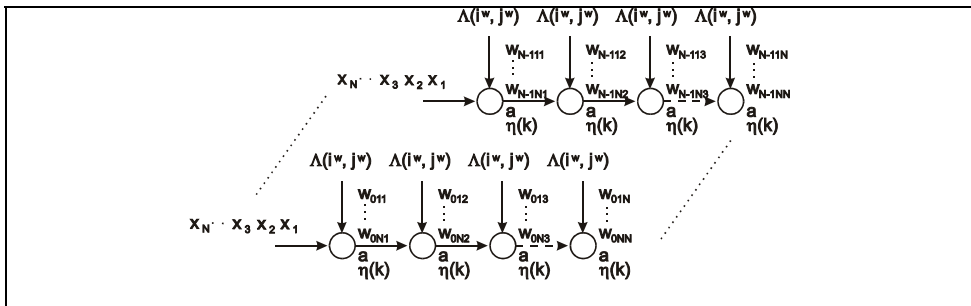


Fig. 8. Systolic array for learning algorithm of Kohonen neural network

8. Efficiency of approach proposed for Kohonen neural network

An efficiency of proposed approach is estimated using the algorithm proposed by Kung (Kung, 1993) and modified for MANTRA computer analysis (Zhang, 1999) (Petkov, 1993). The estimation is based on the dimensions and organization of the Data Dependence Graphs. A computation time for retrieving algorithm equals to:

$$T = (N + K - 1)\tau \quad (22)$$

τ - processing time for elementary processor. The computation time for learning algorithm:

$$T = (N + K - 1)M\eta\tau \quad (23)$$

M - number of learning vectors. Speed-up and processor utilization rate are exactly the same for retrieving and learning algorithms - assuming possible sequential computation time:

$$\text{speed-up} = \frac{NK}{N + K - 1} \quad \text{utilization rate} = \frac{N}{N + K - 1} \quad (24)$$

9. Hamming net description

The Hamming net implements the optimum minimum error classifier when bit errors are random and independent. The performance of the Hamming net is proved in problems such as character recognition, recognition of random patterns and bibliographic retrieval. The feedforward Hamming net maximum likelihood classifier for binary inputs corrupted by noise is presented in (Fig. 9). The Lower Sub Net calculates N minus the Hamming distance to M exemplar patterns, where N is the number of elements in one pattern. The upper sub net selects that node with the maximum output. All nodes use threshold logic nonlinearities, where it is assumed that the outputs of these nonlinearities never saturate. Thresholds and weights in the Maxnet are fixed. All thresholds are set to zero and weights from each node to itself are 1. Weights between nodes are inhibitory with a value of $-\varepsilon$, where $\varepsilon < 1/M$ (Ferrari & Ng, 1992). The connection weights and offsets of the lower sub net are assigned as (Kung, 1993):

$$w_{ij} = \frac{x_i^j}{2} \quad \Theta_j = \frac{N}{2} \quad (25)$$

for $0 \leq i \leq N - 1$ and $0 \leq j \leq M - 1$, Θ_j - the threshold in that node, w_{ij} - the connection weight from input i to node j in the lower sub net. The connection weights in the upper sub net (Maxnet) are fixed as:

$$w_{lk} = \begin{cases} 1 & \text{if } k = l \\ -\varepsilon & \text{if } k \neq l \end{cases} \quad (26)$$

for $0 \leq l, k \leq M$ and $\varepsilon < 1/M$, w_{lk} - the weight from node k to node l in the upper sub net and all thresholds in this max net are kept zero. The outputs of the lower sub net are obtained with unknown input pattern as:

$$\mu_j = \sum_{i=0}^{N-1} w_{ij} x_i - \Theta_j \quad (27)$$

for $0 \leq i \leq N - 1$ and $0 \leq j \leq M - 1$. The inputs of the Maxnet are initialized with threshold logic nonlinearities:

$$y_j^{(0)} = f_i(\mu_j) \quad (28)$$

for $0 \leq j \leq M - 1$. The Maxnet does the maximization:

$$y_j^{(t+1)} = f_i \left(y_j^{(t)} - \varepsilon \sum_{k \neq j} y_k^{(t)} \right) \quad (29)$$

for $0 \leq j, k \leq M$. This process is repeated until convergence. (Kung, 1993)

10. Data Dependence Graph for Hamming neural network

Each node in Data Dependence Graph responsible for Lower Sub Net multiplies the input

signal x_i and corresponding weight w_{ji} which is stored in local memory unit. The product of multiplication is passed to the nearest neighbor on the right hand. The input signals x_i are passed to the nearest bottom neighbors. The μ_i nodes collect the partial products and calculate the global value of coproduct (27). The last $N-1$ columns of nodes on the right are responsible for Maxnet. They transmit step by step the partial results among the neurons which belong to the Maxnet, and they calculate the output value (29) (Mazurkiewicz, 2003b).

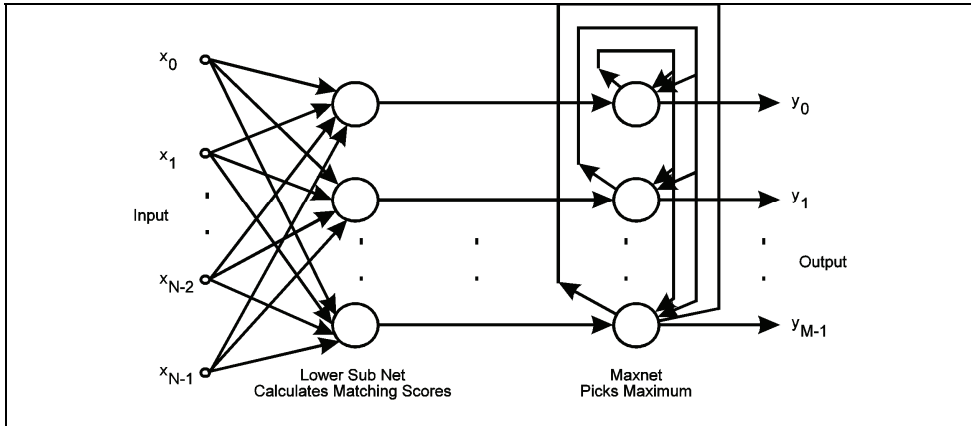


Fig. 9. Hamming neural network

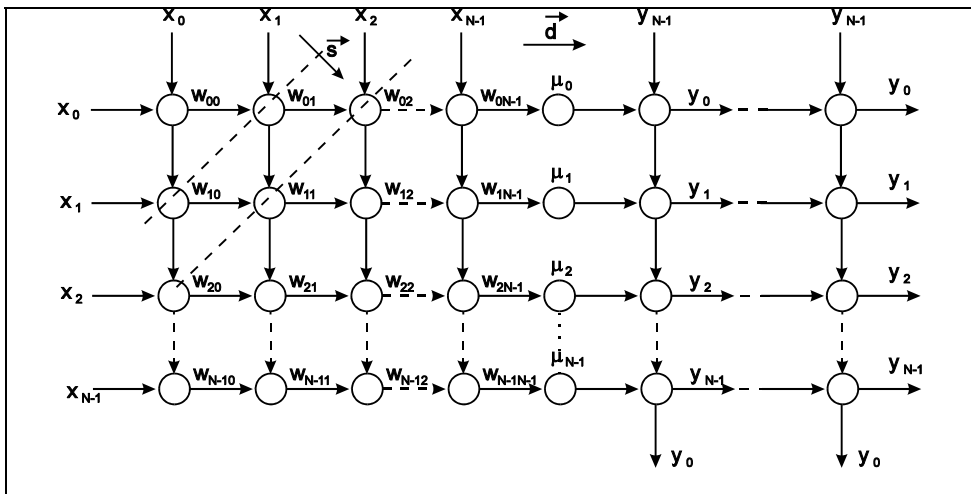


Fig. 10. Data Dependence Graph for Hamming neural network

11. Systolic realisation of Hamming neural network

The Hamming net algorithm is modified for convenience to implement in a parallel systolic architecture. For that the following changes are made: the inputs are considered binary - 0

and 1 - instead of 1 and -1, the weights are made 1 and 0 instead of +0,5 and -0,5 corresponding to a pattern. This will avoid the subtraction of the threshold $\Theta_j = 0,5 * N$ after the summation. The weights are obtained by the following algorithm.

$$w_{ji} = x_i^j \quad \text{for } 0 \leq i \leq N - 1 \quad \text{and } 0 \leq j \leq M - 1 \quad (30)$$

The activation function of the lower sub net is updated as follows:

$$y_j^{(k)} = \begin{cases} y_j^{(k-1)-1} & \text{if } (x_i \oplus w_{ji}) = 1 \\ y_j^{(k-1)} & \text{if } (x_i \oplus w_{ji}) = 0 \end{cases} \quad (31)$$

$y_j = y_j(N)$ for $0 \leq i \leq N - 1$ and $0 \leq j \leq M - 1$ and $0 \leq k \leq N$. It is possible to constitute the maximization network with a layer of up counters, whose outputs are changed with the following algorithm:

$$y_j^{(k)} = \begin{cases} y_j^{(k-1)+1} & \text{if } AND\{O_i\} = 0 \\ y_j^{(k-1)} & \text{if } AND\{O_i\} = 1 \end{cases} \quad (32)$$

O_i - the output vector, obtained from the MSBs - sign bits - of these counters

11.1 Elementary processor realization of Hamming neural net

(Fig. 11) presents the parallel implementation of the algorithm for Hamming net using only digital circuits. The input vector corrupted by noise is applied to the input register when *ILP* - *Input Load Pulse* - is present. The down counter is cleared by this signal. The input data is applied sequentially to each processing element with the *Shift Control Pulse SP'*, which is derived from *SP* and *LCP* - *Load Counter Pulse: not(LCP) & SP*. At the same time the weight register is also shifted once by the same signal. The content of the down counter is updated according to the (8) with the presence of the pulse *CP'*, which is derived from *CP* and *ILP: not(ILP) & CP*.

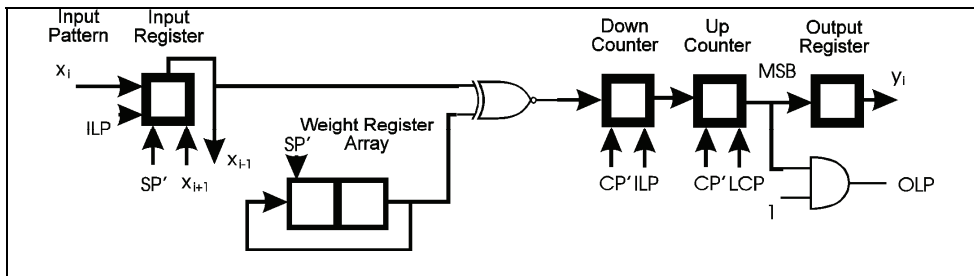


Fig. 11. Single slab implementation of Hamming network

The data obtained in each register after N steps will be negative. They are transferred to the up counters when *LCP* signal is present. *LCP* signal is applied only after N shift operations of the input register and weight registers are performed. The MSBs of the register contents will be the sign bits, which will be 1 for all the data at this stage. The MSBs give the output

vector O_j for $0 \leq j \leq M - 1$. The up counters constitute the Maxnet, which will count up when CP' signal is present. The operation of the up counter is stopped by a control signal OLP - *Output Load Pulse* - derived from the output vector O_j . When any one of the sign bits of the registers becomes 0 (its content becomes positive) OLP stops the operation of the up counter and thereby the neuron with the largest activation function wins the competition. The output pattern is the inverted output of the vector O_j which is obtained when OLP signal is present.

12. Efficiency of Hamming Net Systolic Implementation

12.1 Computation time

The first part of Data Dependence Graph (Fig. 10) is responsible for Lower Sub Net calculation. This part is spread on N elements in horizontal point of view. The second part guarantees the data communication within Maxnet. Here we have $N-1$ elements in horizontal axis. The total computation time for retrieving algorithm taking into account the computation time for Lower Sub Net and the computation time for Maxnet can be calculated as (τ - processing time for elementary processor):

$$T_{systol} = (4N - 3)\tau \quad (33)$$

12.2 Processor utilization rate

If we assume that all elementary processors need the same time-period to proceed their calculations the speed-up and utilization rate factors can be estimated as:

$$speed - up = \frac{(2N - 1)N}{4N - 3} \quad utilization\ rate = \frac{(2N - 1)}{4N - 3} \quad (34)$$

13. Conclusion

The comparison of the same criteria for two methods of learning is the most interesting part. Computation Time - if we assume single presentation of each training vector - is less then two times longer for Delta-rule learning. Of course such assumption is true for Hebbian learning but isn't in general true for Delta-rule. Each next presentation of training set makes the Computation Time longer and the dependence is directly proportional.

It is very interesting we can observe exactly the same Speed-Up and Processor Utilization Rate both for Hebbian and Delta-rule learning procedures. The necessary time-period for calculation of Delta-rule procedure is longer than time-period related to Hebbian learning - but elementary processors' using is the same. The results of discussion show that it is possible to create the universal structure to implement all algorithms related to Hopfield neural network. This way there are no barriers to tune the Hopfield net to completely new tasks. The proposed methodology can be used as a basis for VLSI structures which implement Hopfield net or as a basis for set of general purpose processors - as transputers or DSP (Shiva, 1996). Proposed methodology for Kohonen neural is based on classical and not modified algorithms related to Kohonen maps. It is possible to realize the obtained subtasks by software processes, but also using dedicated neuro-computers like MANTRA (Zhang, 1999) (Petkov, 1993).

The Hamming net algorithm is modified suitably for the systolic implementation. The systolic algorithm for the computation of the activation values of the lower sub net is developed. The upper sub net which acts as a maximization network is constructed by using simple counters. The main advantage of the proposed architecture is that it can be easily extended to larger networks.

The minimum values of the efficiency parameters are calculated for the proposed structure with single available (not-failed) processor. The maximum values are related to the optimal number of used processors. This way it is possible to observe the changes of the values as a function of ready to use elementary processors. We can model the influence of the decrease of the number of processors for the global efficiency of the system.

	Learning		Retrieving phase
	Hebbian rule	Delta-rule	
Computation time T_{systol} (min)/(max)	$3(2N-1)\tau M / 3N^2\tau M$	$5(2N-1)\tau M\beta / 5N^2\tau M\beta$	$2(2N+1)\tau\beta / 2N(N+2)\tau\beta$
Speed-up (min)/(max)	$1 / \frac{N^2}{2N-1}$	$1 / \frac{N^2}{2N-1}$	$1 / \frac{(N+2)N}{2N+1}$
Utilization rate (min)/(max)	$\frac{N}{2N-1} / 1$	$\frac{N}{2N-1} / 1$	$\frac{N+2}{2N+1} / 1$

Table 1. Efficiency parameters for ring systolic structure related to Hopfield neural network algorithms – possible minimum and maximum values

14. References

- Asari, K.V. & Eswaran, C. (1992). *Systolic Array Implementation of Artificial Neural Networks*, Indian Institute of Technology, Madras
- Ferrari, A. & Ng, Y.H. (1992). A Parallel Architecture for Neural Networks, *Parallel Computing '91*, pp. 283–290, Elsevier Science Publishers B. V.
- Kung, S.Y. (1993). *Digital Neural Networks*, PTR Prentice Hall
- Mazurkiewicz, J. (2004). Feedforward Neural Network Simulation Based on Systolic Array Approach, *NETSS 2004*, pp. 17-22, ACTA MOSIS No. 94, MARQ., Ostrava
- Mazurkiewicz, J. (2005a). Kohonen Neural Network Learning Algorithm Simulation Based on Systolic Array Approach, *MOSIS'05 Conference Modelling and Simulation of Systems*, pp. 202-207, ACTA MOSIS No. 102, Ostrava
- Mazurkiewicz, J. (2005b). Systolic Realization of Kohonen Neural Network, *Artificial Neural Networks: Formal Models and Their Applications – ICANN 2005*, pp. 1015-1020, LNCS 3697, Springer-Verlag Berlin Heidelberg
- Mazurkiewicz, J. (2003a) Systolic Realisation of Self-Organising Neural Networks, *ICSC 2003*, pp. 116-123, EPI Kunovice
- Mazurkiewicz, J. (2003b). Systolic Simulation of Hamming Neural Network, *Advances in Soft Computing*, pp. 867-872, Physica-Verlag Heidelberg, A Springer-Verlag Company
- Petkov, N. (1993). *Systolic Parallel Processing*, North-Holland
- Shiva, S.G. (1996). *Pipelined and Parallel Computer Architectures*, Harper Collins Publishers
- Zhang, D. (1999). *Parallel VLSI Neural System Design*, Springer-Verlag

Smart RFID Security, Privacy and Authentication

Mouza A. Bani Shemali, Chan Yeob Yeun and Mohamed Jamal Zemerly
*Khalifa University for Science, Technology and Research
 United Arab Emirates*

1. Introduction

Radio-frequency identification (RFID) is an automatic identification method, used to transmit the identity such as serial number of objects or subjects (people) wirelessly, through radio waves. RFID technology is a new promising technology that will spread in the near future to enter most of our everyday activities.

An RFID system consists of three main components; a tag, a reader, and a server. There are three types of tags as follow:

1. Passive tag

Passive tags need to be beamed by the reader to be activated. Passive tags are also smaller, less expensive than other kind of tags and used for a short range.

2. Semi Passive tag

Semi passive tags have an on-board power source to run the tag chip circuit and draw the communication energy from the reader. Besides, semi passive tags have longer read range than the passive tags.

3. Active tag

Active tags include miniature batteries used to power the tag, so RFID reader can read active tags at distances of one hundred feet or more. Also, active tags can be used as sensors and are more expensive than other kind of tags. Table 1 shows the advantages and disadvantages of the three types of RFID tags.

Tag Type	Advantages	Disadvantages
Passive	<ul style="list-style-type: none"> • Longer Life time • Lowest Cost • More Flexible 	<ul style="list-style-type: none"> • Distance Limited
Semi Passive	<ul style="list-style-type: none"> • Longer range for Communication • Can be used as sensors 	<ul style="list-style-type: none"> • Expensive due to the battery • Cannot determine if the battery is good or bad
Active		

Table 1. Comparison of various types of tags

Some smart tags have memories that can be written into and erased, while others have memories that can only be read, so the cost of the tag depends on the memory size that it contains.

As for the reader it consists of two parts. First part is an antenna which is used for communication with RFID tags wirelessly. Second part is an electronic module that is networked to the host computer through cables and relays messages between host and all tags within antenna's range. Also, the electronic module is responsible of some security functions such as encryption/decryption, and authentication. The last part is the server (a PC or a workstation) which is considered as the brain of an RFID system. The server is responsible of tracking movement and redirecting the objects through the system and verifying identity and granting authorization for the tags. The RFID system communication starts when the reader emits radio waves to query the tag, then the tag transmits its stored data to the reader which will relay the tag's data back to the server. The server is responsible of the following tasks:

1. Controls the system's data purchase.
2. Keeps inventory and alerts suppliers when new inventory is needed.
3. Tracks movement and redirects the objects through the system communication.
4. Verifies the identity of tags and grants authorizations for them. Figure 1 shows the RFID system communication.

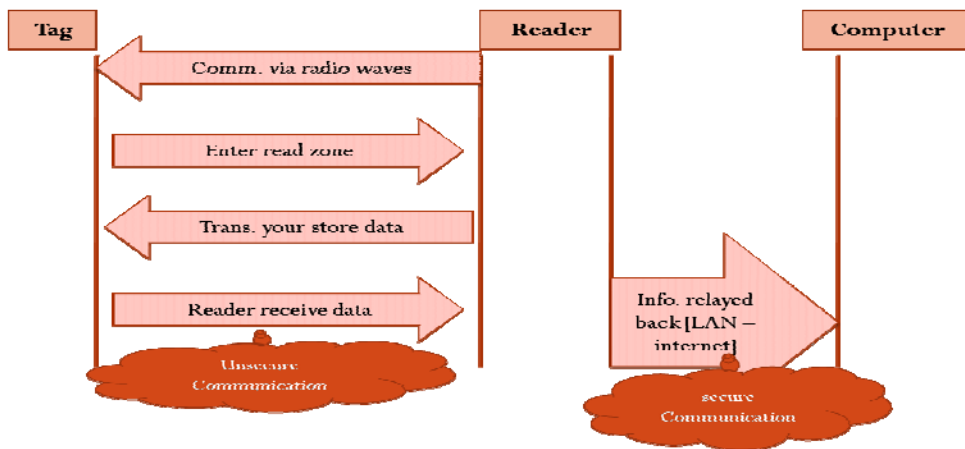


Fig. 1. RFID system communication

RFID technology is predicted to be a substitute for the second generation of the barcode technology since there are four main advantages for the RFID technology over barcode (Hunt *et al.*, 2007) as follows:

1. RFID eliminates the need for direct line-of-sight reading that the barcode depends on.
2. RFID scanning can be done at greater distances than the barcode scanning.
3. RFID can scan multiple products simultaneously.
4. Since RFID can be used as a unique system identifier and can be used as a product pointer in the database, which can facilitate the tracking of all products history.

Most RFID applications today utilize the passive tags as they are so much cheaper to manufacture and operate over four ranges of frequency. Table 2 shows the comparison between the four different types. The antenna shape is also important to the tag's performance as the larger the antenna, the more energy it can collect.

Microwave 2.45GHz & 5.8GHz	Ultra-High Frequency (UHF) 868- 915 MHz	High Frequency (HF) 13.56 MHz	Low Frequency (LF) 125 KHz	Frequency Range
Longest	Medium	Short	Shortest	Typical Max Read Range
Fastest	Fast	Moderate	Slower	Data Rate
Worse	Poor	Moderate	Better	Ability to read near metal or wet surfaces

Table 2. Comparison between the four frequencies type

Recently, RFID technology can be considered as the niche development technology. However, they have limited power constraint (powerless for passive tags), limited communication range, and a small number of gates for logical operation. All of these limitations led to building RFID systems but without a security aspect. As a result RFID technology now faces some major security issues that may hinder its propagation if not handled properly. In this chapter we will focus on the passive RFID tags and its security development. The rest of this chapter is organized as follows. RFID application examples are pointed out in Section 2. The security challenges and the practical secure implementation of RFID in general and related work are summarized in Sections 3 and 4. The analysis of some of the privacy and authentication solutions are given in Section 5. We describe applications of Smart E-Travel based on RFID in Section 6. We conclude this chapter with future work in Section 7.

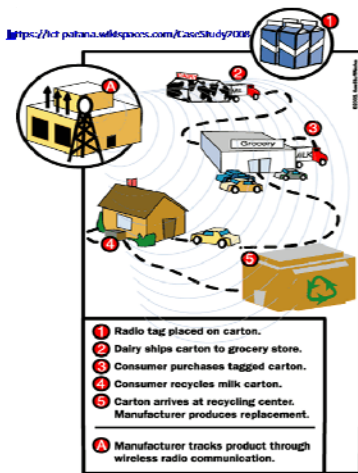
2. Application Areas

RFID technology is used anywhere that needs a unique identification system, hence the RFID system able to identify the objects or the subjects by means of the serial number. Thousands of companies worldwide have resorted to RFID systems to improve efficiency in production and to automate routine decision-making. Because, RFID tags can automate the computers to do the next steps, without human interference (Henrici, 2008).

Therefore, RFID applications are widespread nowadays; here we will introduce only some of them as follow:

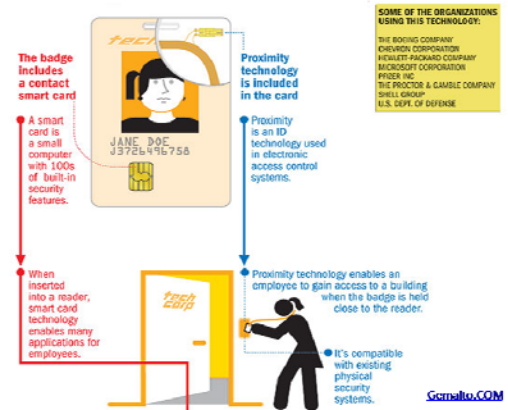
- **Product tracking:** tracking goods in the supply chain and during the manufacturing process by using Electronic Product Code (EPC) which can provide a unique ID for any physical object.
- **Building access:** allows controlled access to buildings and networks.

- **Human implantation:** implanting RFID tag in the human body, so it can be used for information storage, including personal identification, medical history, medications, allergies, and contact information related to the person with the tag.
- **Hospital:** using RFID for patient identification and portable asset tracking.
- **Libraries:** list a lot of library items in their collections in a short time. To allow users to automatically check out and return library property. Besides speeding up checkouts, keeping collections in better order, RFID provides a better control on theft, non-returns, and misfiling of a library's assets.
- **Transportation:** using RFID in toll collection, ticketing, vehicles tracking, e-Passport, RFID baggage sorting system, and other transport applications. Figure 2 shows some examples of the RFID applications. In Section 6 we give a scenario that shows how RFID technology applies in the airport environment.

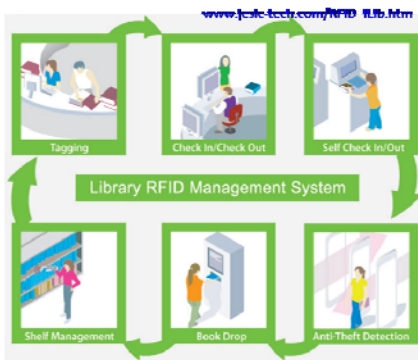


Product Tracking

A smart employee badge contains both smart card and proximity technologies.



Building access



RFID in the Libraries



Transportation

Fig. 2. Examples of RFID applications

3. Security Challenges and Threats

Although the use of RFID tags continues to increase according to a new report from In-Stat (Instat, 2009), which states that over 1 billion tags were produced last year, and by 2010, the number will rise to **33 billion**. However, the RFID technology faces some security threats. Threats are potential events that cause a system to respond in an unexpected way e.g. attacker attacks the system causing some damage to the RFID system (Henrici *et al.*, 2009). Threats to RFID system are categorized into seven threats that are listed below.

1. **Spoofing identity.** Occurs when an attacker successfully gains an unauthorized access to the RFID system. Any attacker with suitable equipment is able to clone any legitimate tag and communicate with a legitimate reader as a genuine tag where in fact it is a fake tag. Figure3 illustrates how adversary could clone RFID tag.



Fig. 3. Process of cloning RFID tag

2. **Tampering with data.** This deals with the tag integrity and occurs when an attacker modifies, adds, deletes, or reorders data in the RFID tag.
3. **Repudiation.** Occurs when a user denies an action that was performed during the execution of the RFID protocol.
4. **Privacy disclosure.** Occurs when information is exposed to an unauthorized user and that can cause some privacy violation. Actually, there have been issues which have arisen from by privacy advocates over the use of RFID tags to track people or their tagged stuff. Also, a tag emits data to any reader without alerting its owner. This can be made worse if the tag contains some personal data such as name, birthday, etc. related to the tag owner so the attacker will not only be able to track the tag owner but also he/she could create a profile that relates to that person. Figure 4 depicts one of the privacy disclosure issues in the RFID.
5. **Denial of service.** This deals with the availability of the tag data and occurs when an attacker denies service to valid users. In the RFID system it occurs when an attacker prevents the tag to update a value after a successful authentication in the RFID protocol.

6. **Elevation of privilege.** Occurs when an unprivileged user can gain a higher privilege in the RFID system than what they are authorized for.
7. **Man-in-the-Middle attack.** Occurs when the attacker creates a connection between the legitimate reader and tag and through this connection the attacker is able to catch the messages between the reader and the tags or even interrupt and modify these messages.

Researchers are currently seeking solutions to solve the security issues in RFID, so it can be proliferated without any shortcomings in the future. In the next section we divide the research work into two major areas.

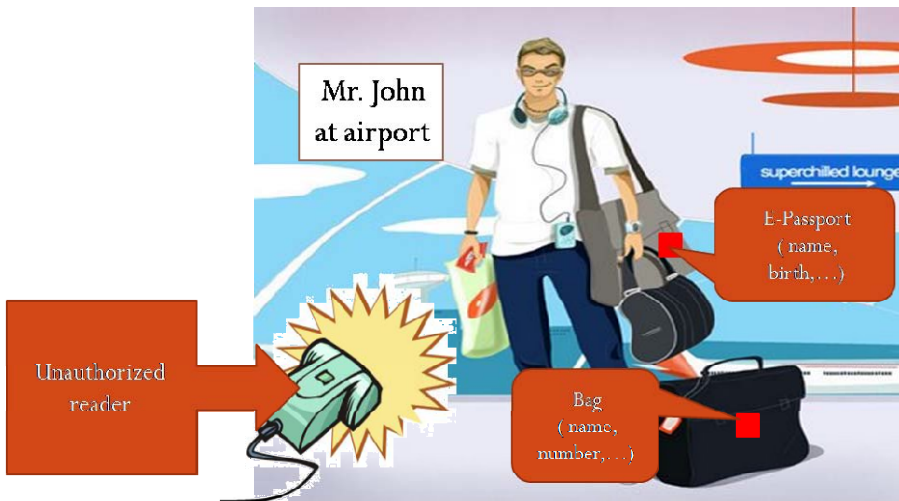


Fig. 4. Security issues- profiling process

4. Practical secure implementation

In order to build RFID security algorithms, the designer first must understand the applications that RFID systems will be used in, so that he/she can design algorithms with suitable security approach. We can classify the practical security approach into three types (Karygiannies *et al.*, 2008) as follow:

1. Randomization Approach

Random assignment schemes depend on the random number or pseudorandom rotation to create challenges in order to hide the real tag identifiers. We can divide Randomization approach into two types as follow:

a- Pseudorandom Rotation

Create a list of pseudorandom numbers that are stored in both of the tag and the reader memory. Such approach is like Juels (Juels, 2004) proposed, he

suggested the idea of Minimalist cryptography which consists of storing a short list of random pseudonyms in the tag so each time a tag is queried it emits the next pseudonym in the list until the end of the list. Then it starts from the beginning until it ends. This scheme can be implemented in an RFID tag by just adding several hundred bits of memory to the tag with enabling the read write feature. Using this mechanism helps to prevent the tracking of the tag by illegitimate reader.

Also, (Peris-Lopez *et al.*, 2006) proposed a lightweight mutual authentication protocol based on the idea of Minimalist and index-pseudonyms (IDSs). Each tag stored key is divided into four parts of 96 bits ($K = K1 || K2 || K3 || K4$), $X || Y$ denotes the concatenation of data items X and Y and they are updated after each successful authentication. This protocol consists of four steps. Tag Identification, Mutual Authentication, Pseudonym Index Updating, and Key Updating. Also, pseudorandom can be generated by implementing a hardware device in RFID tags that present some of the challenge response authentication protocol between tag and reader.

One of the algorithms that used hardware approach is (Lee & Hong, 2006) algorithm that used a pseudorandom pattern generator (PRNG) implementation using linear feedback shift registers with self-shrinking generator (SSG). This algorithm is used to authenticate the tag to the reader by exchanging a challenge-response using SSG.

b- **Random Number**

In the random number approach the tag identifies itself with random identifier that is not related to its serial number such as Meta-id. An example of this approach is (Lee & Verbauwhede, 2005) algorithm which is a lightweight authentication protocol that can be used for low cost RFID called Advanced Semi-Randomized Access Control (A-SRAC). First of all, a reader sends a query and a random number R_s to the tag. Then, the tag will generate a random number R_t and send it to the reader with the tag MetaID. After that, the reader relays this message back to the server through a secure channel.

The server looks up the key corresponding to the tag MetaID, then the server will check the uniqueness of the MetaID among other MetaIDs in the system. If that MetaID is not unique then the server will generate random number R_2 till it reaches the new unique MetaID. Then, the server will send R_2 and $h(\text{key} || R_2 || R_1)$ to the tag through the reader. The tag will check the correctness of the message if it is correct then, it will update the previous key with the new key.

2. **Encryption Approach**

Encrypt the data between the reader and the tags. The tags only store the ciphertext and are not responsible of any encrypt or decrypt operation. The encrypt/decrypt operation are done by reader or other enterprise subsystem components. We can divide the encryption approach into two parts as follow:

a- Secret Key Cryptosystem

This approach needs a shared secret key between reader and tag to encrypt the messages between the reader and the tag. The use of secret key cryptosystem approach can provide either one way or mutual authentication mechanism between the reader and the tag. The Randomized Hash Locks algorithm (Weis *et al.*, 2004) is a lightweight authentication algorithm that can be embedded into low cost RFID tags. Randomized Hash Locks is a scheme for mutual authentication between RFID reader and tag. A reader contains a list of the tags keys and each tag stores its own key. In the first step, a reader sends "Who are you?" message to the tag. Then, the tag will generate a random number R and sends it along with the hash value of the tag stored key. When the reader receives the tag message it will start to compute the hash value for every key in the list and compares it with the tag message. Finally, after finding the corresponding key from the comparison then the reader will send "You must be K " message, which K is the tag identifier, to the tag so the tag will make sure that the reader is a valid one.

Also, (Ilic *et al.*, 2008) proposed the Synchronization Approach as a solution to authentication issues in the RFID system. The scheme verifies and updates the synchronized secrets of tags. It states that each tag has a secret K_x shared in both of the reader and tag memory on every communication between reader and tag. After each communication the secret K_x will be updated between both of the reader and the tag, so it will increase by one and the secret will be K_{x+1} . Therefore, if a genuine tag identifier and the synchronized secret are copied to a fake tag, and by then the fake tag tries to interrogate with the reader, the reader will be able to detect de-synchronization. This approach is cost-effective and can be implemented in low cost RFID tags.

Moreover, (Song & Mitchell, 2008) proposed a protocol which consists of three exchanges between the reader and the tag. Each tag stores a hash value of string μ [$t = h(\mu)$] unique to each tag. Also, each server stores $[(\mu, t)_{new}, (\mu, t)_{old}, D]$ where $(\mu, t)_{new}$ is the new values of the string μ and corresponding $h(\mu) = t$, and $(\mu, t)_{old}$ is the previous stored data, and D is the data of the tag such as price. After a successful authentication both the server and tag will update their values.

Furthermore, (Lu *et al.*, 2007) suggested the Key-Updating scheme to solve the problem of keys compromised in tree approach scheme (Molnar *et al.*, 2005) which states that a temporary key is used to store the old key for each non-leaf node in the key tree. For each non-leaf node, a number of state bits are used in order to record the key-updating status of nodes in the sub-trees such as 1 bit for having been updated, otherwise it will have 0 bit. Based on this design, each non-leaf node will automatically perform key-updating when all its children nodes have updated their keys.

Another algorithm that is considered one of the most secure classic algorithms is the One Time Pad (OTP) (Stallings, 2002). Typically, the pad is generated in some random way and is shared between the senders and the receivers. Usually, the key will expire as soon as it has been used once. When a message is to be sent, the sender uses the secret key to encrypt each character, one at a time. The encryption algorithm is simply the XOR operation between the message and the key. Only the sender and receiver have the ability to encrypt and decrypt the message using the

shared secret pads. Once the one-time pad is used, it cannot be reused. If it is reused, someone who intercepts multiple messages can begin to compare them for similar coding for words that may possibly occur in both messages. This algorithm is simple and can be used to secure the insecure RFID communication. However, OTP has some disadvantages such as having long messages requires long keys. Also, distributing the pad in a secure manner is difficult.

b- Public key Cryptosystem

This approach is similar to the secret key approach but uses two keys: public key and private key. The public key is used to convert from plaintext to ciphertext which will be stored in the tag memory. After that only the holder of the private key will be able to decrypt the ciphertext stored on the tag. The agency that holds the private key must be a trusted agency. An example that uses this approach with a trusted agency is the re-encryption approach that uses the European Central Bank as a trusted third party.

When the European Central Bank proposed using RFID in banknotes (Juels & Pappu, 2003) proposed a re-encryption scheme to solve the privacy issues in the RFID. Re-encryption is changing the appearance of the ciphertext without changing the plaintext. The re-encryption schemes may be done by shops, banks, or by consumers that hold the banknotes. An RFID banknote has a memory that has a serial number, a signature, a cipher text, and a random number which are used in the El-Gamal algorithm (El Gamal, 1985) that is used to re-encrypt the ciphertext and save it in the RFID tags. The drawback of this algorithm is that the re-encryption algorithm may not be done frequently enough.

Universal re-encryption suggested by (Golle *et al.*, 2004) is another algorithm that uses the Public key Cryptosystem. Universal re-encryption is a cryptographic technique that is similar to the El-Gamal cryptosystem except that it does not require a public key. In the universal re-encryption the input plaintext must be encrypted by the recipient public key before it enters the mix servers that consist of a chain of involved servers. Each server involved in the scheme re-encrypts the input ciphertext from the previous server until it reaches the last sever so the recipient should have the whole output ciphertext from the mixnet servers then decrypts them all using his/her private key until it has the match cipher that is encrypted under his/her public key.

This scheme can be used to enhance privacy in RFID tags so they can be re-encrypted under the agency that generates them. For example, a man walking home with his bag that has an RFID tag which can be re-encrypted by the stores related to that bag all along the way to the man's house. Universal re-encryption may be an efficient scheme but it has some limitations such as the recipient should decrypt all the output cipher text to have his/her plaintext.

Next section analyzes the previously mentioned algorithms; however most of the algorithms actually focus either on the authentication or privacy issues. Therefore, in the next section we divided the analysis of the algorithms into two parts: privacy solutions analysis, and authentication solutions analysis.

5. Analysis of the privacy and authentication solutions

We can evaluate the RFID privacy and authentication algorithms based on four categories as the following:

1. **Cost and Complexity:** which depend on the memory size of the RFID tags and the number of gates that may be needed for the algorithm, adding these to the tag can affect the tag cost. Since RFID tags are used in large scale, so the cost of the tag depends on how much of the resources the algorithm will use, so we need to reduce the size of the memory needed for the algorithm and the number of the gates. Therefore, in designing RFID algorithms there is a need to try to create a simple algorithm that needs a small amount of memory and gates. For example, for the SHA-1 algorithm about 4200 gates are required where lighter hash algorithm need about 1700 gates (Yu *et al.*, 2004) which makes this algorithm less complex than the SHA-1 algorithm.
2. **Performance:** we can measure the performance of the RFID algorithm by estimating the times of each message round trip, the time to retrieve the data from backend server, and to read and write the data to the tag. So the performance can be improved by reducing the size and the number of the messages in the algorithm. For example, public key algorithms are slower than secret key algorithms, so the public key algorithm must be used in case the message is so small, but if we have large amount of data then secret key algorithm will perform better.
3. **Availability:** the RFID system is used in critical businesses that the system must be available all the time such as using RFID in a supply chain. Therefore, the availability of the system must be guaranteed during the execution of the algorithm.
4. **Anonymity:** Tags must have anonymity to prevent the tracking problem. So, the tag response must appear as a random number and refreshed frequently so the attacker will not be able to trace.

This subsection uses the four categories that are mentioned above to evaluate the RFID security algorithms. Here is the analysis of each algorithm after dividing the algorithms into two parts as follow:

5.1 Privacy Solutions Analysis

1. **Minimalist:** (Juels, 2004) points out that needs more memory to store the list of pseudonyms and the communication cost per session will be a little bit costly. On the other hand, the performance of the protocol is good since the computation in the tag side will be limited to some string comparisons and XOR operation. Furthermore, Since Minimalist list use two exchange identifiers and refresh the pseudonyms after each successful authentication in the protocol it will help to prevent the denial of service attack and tracking problem so the algorithm can provide the availability and the anonymity features.

2. **Re-encryption scheme and universal re-encryption:** Since re-encryption and universal re-encryption schemes use public key cryptographic scheme then the cost of the algorithms will be costly since these schemes need more memory. The complexity of these algorithms increases with the number of logic gates. Moreover, these algorithms will need a lot of computation on the server side which will lead to slower response from the server side and will take time to write the cipher text on the tag chip so the performance of the algorithms will be affected. However, these algorithms can provide anonymity to the tag identifier by re-encryption scheme.

5.2 Authentication Solutions Analysis

1. **Peris-Lopez Algorithm:** Since this algorithm uses as a basis the Minimalist algorithm then the algorithm will face the same conditions, so the algorithm will be a little bit costly. Moreover, this algorithm does not need a lot of the computation power by either of the tag or the server side so it will perform very well. However, the algorithm faces Desynchronization attack (Li & Wang, 2007) so the availability of the system cannot be ensured all the time. Finally, the algorithm uses four keys to ensure the anonymity of the identifier.
2. **SSG Algorithm:** this is a low cost and simple algorithm that uses small size of memory with a small number of gates compared with other security algorithms. Moreover, the messages of this algorithm can transfer quickly between the tag and the server so the performance of the algorithm is good. But, the algorithm is vulnerable to Desynchronization attack so the availability of the data is an issue here. Finally, the algorithm is able to change the identifier after each successful authentication.
3. **A-SRAC:** the algorithm uses simple computation so it is not that costly or not a complex one. Moreover, the number of the sent messages and the size of the messages are small so the algorithm performance is good. Also, the server saves the old and the new data to prevent Denial of Service (DoS) attack. Also, the algorithm uses Meta- id to prevent the tracking of the tag.
4. **Randomized Hash Locks:** heavy weight solution if the key list is long and it could be costly. Besides, the algorithm is not resistance to DoS attacks. Yet, the algorithm is able to prevent tag trace identifier using hash algorithm.
5. **Synchronization Approach:** simple and does not need a lot of memory size so the algorithm is cheap with low complexity. However, the algorithm is vulnerable to Desynchronization attacks. Finally, the algorithm is able to provide anonymity since the identifier is different in each session.
6. **Song and Mitchell Algorithm:** this algorithm uses simple computation and little memory size so it is simple with low complexity. In addition, since the server stores the old and the new values of the tag identifiers then the algorithm can

provide better availability. However, the algorithm is vulnerable to face attacks that prevent the anonymity of the tag identifier.

7. **Lu et al. Algorithm:** this algorithm needs a lot of communication and comparison on server side so its performance is not that good. However, the algorithm is able to prevent DoS attacks and maintains the anonymity of the tag by using more than one key for each tag. Table 3 shows a summary of the algorithms with their evaluation.

In the next section we present an application of how to implement RFID in the airport considering the security aspects that were previously mentioned. Moreover, we think that the real problem in most of this technology implementation is that it is applied without considering the security threats. Therefore, the real challenge can be how to apply the RFID technology in a safeguard way.

Algorithm	Algorithm Solves		Cost Complexity	Performance	Availability	Anonymity
	Privacy	Authentication				
Minimalist	√		X	√	√	√
Peris-Lopez Algorithm		√	X	√	X	√
SSG Algorithm		√	√	√	X	√
A-SRAC		√	√	√	√	√
Randomized Hash Locks		√	X	X	X	√
Synchronization Approach		√	√	√	X	√
Song and Mitchell Algorithm		√	√	√	√	X
Lu et al. Algorithm		√	X	X	√	√
re-encryption scheme	√		X	X	N/A	√
universal re-encryption	√		X	X	N/A	√

Table 3. Summary of the proposed algorithms with their evaluation

6. Smart e-Travel Scenario

First of all, we aim to use e-passport with RFID chip in a secure manner. We will focus here on the authentication issues. In the beginning, each e-passport holder must authenticate himself to the e-passport issuer authority by providing his picture and fingerprint to adjust this data to each passport holder in a safe database. Then the E-passport tag will only contain the encrypted password of the matched tag holder data in the database. Now we can state the e-passport holder application in the airport as follow:

The passenger holds his e-passport (e-passport contains encrypted data that identifies the passenger) and enters the airport. At the check-in point, there is a reader that reads the encrypted data in the passenger e-passport then it will match this data to the data in the back end database. If there is a match then the passenger is considered as an authorized person in the airport environment, and can enjoy the facilities of the airport. After the authentication process the reader will ask for the mobile number of the passenger, the passenger mobile must be able to read RFID tags (Evans, 2005). Then some applications will be downloaded to the passenger mobile so now he/she can tour in the airport easily using this application. The application can read the tags in the airport and show the passenger the airport facilities such as airport bathroom, or coffee shops. Moreover, when the passenger luggage which contains an RFID tag to facilitate luggage tracking securely reaches the flight then, the passenger will be notified by sending an SMS message to his/her mobile to inform him/her about the place of the luggage.

Of course, this scenario needs security features, so that nobody except the airport authority can read the passenger personal data. Also, no unauthorized person can fool the airport reader and enters the airport as an authorized one.

7. Future Work

The scenario described in Section 6 requires a lightweight mutual authentication protocol that meets all the evaluation categories. Moreover, the proposed solution needs to have low cost and can easily be implemented. It is envisaged that the algorithm will be based on the Shrinking Generator (SG) mechanism. The idea of SG can be used to provide cryptographic services to secure the communication over unsecured channels and it is suitable to be implemented in RFID system.

In summary, the RFID systems are emerging technologies that will propagate in our daily life in the future. However, these technologies have some security concerns especially in the privacy and authentication issues. This chapter reviewed some of the proposed solutions to solve the privacy and authentication issues in the RFID. Also, this chapter analyzed the proposed solutions upon some evaluation categories. In the end, the chapter shows a scenario of how to implement the RFID technology in the airport in security manner.

8. References

- El Gamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms, *Proceedings of CRYPTO 84 on Advances in cryptology*, Santa Barbara, California, United States, Springer-Verlag New York, pp. 10-18.
- Evans, M. (2005). Prototype Nokia 3220 NFC RFID phone could reshape society, Mobile Mentalism.com, <http://mobilementalism.com/2005/12/12/prototype-nokia-3220-nfc-rfid-phone-could-reshape-society>
- Golle, P. ; Jakobsson, M., Juels, A. & Syverson, P. (2004). "Universal encryption for Mixnets", *Proceedings of CT-RSA 2004*, In T. Okamoto (Ed.), LNCS 2964, pp. 163-178.
- Henrici D. (2008). *RFID Security and Privacy Concepts, Protocols, and Architectures*, Springer.
- Henrici, D.; Kabzeva, A., & Mueller, P. (2009). RFID System Architecture Reconsidered, In: *Development and Implementation of RFID Technology*, Turcu, C. (Ed.), In-Tech.
- Ilic, A.; Lehtonen, M., Michahelles, F. & Fleisch, E. (2008). Synchronized Secrets Approach for RFID- enabled Anti-Counterfeiting, *Proceedings of Demo at Internet of Things Conference 2008*, Zurich, Switzerland.
- Instat (2009). RFID Tag Market to Approach \$3 billion in 2009, <http://www.instat.com/newmk.asp?ID=1206>
- Juels, A. (2004). Minimalist cryptography for low-cost RFID tags, *Proceedings of Int. Conference on Security in Communication Networks - SCN 2004*, LNCS 3352, Amalfi, Italy, Springer-Verlag, pp. 149-164.
- Juels, A. & Pappu, R. (2003). Squealing Euros: Privacy-protection in RFID-enabled banknotes, *Proceedings of Financial Cryptography*, Gosier (Ed.), Guadeloupe, FWI, LNCS 2742, Springer-Verlag, pp.103-121.
- Karygiannies, A.; Eydt, B., Phillips & Ted S. (2008). Practical Steps for Securing RFID Systems, In: *RFID Handbook Applications, Technology, Security, and Privacy*, Ahson, S. & Ilyas, M. (Ed.), Taylor & Francis Group.
- Lee, H. & Hong, D. (2006), The tag authentication scheme using self-shrinking generator on RFID system, *World Academy of Science, Engineering and Technology* Vol. 18 , pp. 52-57.
- Lee, K. & Verbauwhede, I. (2005). Secure and Low-cost RFID Authentication Protocols, *Proceedings of the 2nd IEEE International Workshop on Adaptive Wireless Networks (AWiN)*.
- Lu, L.; Han, J., Hu, L., Liu, Y. & Ni, L. (2007). Dynamic Key-Updating: Privacy-Preserving Authentication for RFID Systems, *Proceedings of Pervasive Computing and Communications*, pp. 13-22.
- Li, T. & Wang, G. (2007). Security Analysis of Two Ultra-Lightweight RFID Authentication Protocols, *Proceedings of the 22nd IFIP TC-11 Int'l Information Security Conference*, Vol. 232, Springer, pp. 109-120.
- Molnar, D.; Soppera, A. & Wagner, D. (2006). A Scalable, Delegatable Pseudonym Protocol Enabling Ownership Transfer of RFID Tags, *Proceedings of SAC*, LNCS 3897, Springer-Verlag, pp. 276-290.
- Peris-Lopez, P.; Hernandez-Castro, J., Estevez-Tapiador, J. & Ribagorda, A. (2006). EMAP: An Efficient Mutual-Authentication Protocol for Low-Cost RFID Tags, *Proceedings of OTM Federated Conference and Workshop: IS Workshop*, pp. 352-361.
- Hunt, V.; Puglia, A. & Puglia, M. (2007). *RFID: A Guide to Radio Frequency Identification*, Wiley-Interscience, April 10.

- Song, B., & Mitchell, C.J. (2008). RFID authentication protocol for low-cost tags, *Proceedings of the First ACM Conference on Wireless Network Security 2008*, Gligor, V. D.; Hubaux, J. P. & Poovendran, R. (Ed.) , Alexandria, VA, USA, March 31 - April 02, 2008, pp.140-147.
- Stallings, W. (2002), *Cryptography and Network Security*, Third Edition, Prentice Hall.
- Weis, S.; Sarma, S., Rivest, R. L. & Engels, D. W. (2004). Security and Privacy Aspects of Low-cost Radio Frequency Identification Systems, *Proceedings of Security in Pervasive Computing*, LNCS 2802, pp. 201-212.
- Yu, M.; Zhou, T., Wang, J. & Ye, Y. (2004). An Efficient ASIC Implementation Of SHA-1 Engine For TPM, *Proceedings of IEEE Asia-Pacific Conference on Circuits and Systems*, pp. 873-876.

Security and Privacy of Intelligent VANETs

Mahmoud Al-Qutayri, Chan Yeun and Faisal Al-Hawi
*Khalifa University of Science, Technology and Research
United Arab Emirates (UAE)*

1. Introduction

The rapid advancement and pervasiveness of wireless communications and information technologies are revolutionizing many aspects of the human lifestyle. The convergence of these technologies is enabling the delivery of a wide range of services and applications of personnel as well as public nature. An application area which is expected to benefit greatly from this is advanced vehicle safety. Car manufacturers have started incorporating some of the wireless information communication technologies (ICT) in their cars with applications covering safety, traffic efficiency, driver assistance, and infotainment. They are utilizing the dedicated short range communication (DSRC) to deliver these applications (Eichler, 2007). The goal is to have fully integrated Intelligent Transportation Systems (ITS) that increase the overall safety and efficiency of transportation in the future.

Smart vehicles with the appropriate wireless ICT will in the near future be able to communicate with each other as well as road-side units (RSUs) located at key points on the road, such as junctions. This enables the formation of self-organized networks connecting the vehicles and RSUs. The RSUs can also be connected to a backbone network if needed. This new form of networks is called VANETs (Vehicular Ad-hoc NETWORKs). In VANETs, the vehicles or RSUs nodes act both as end points and routers. Due to their ad-hoc mobile nature VANETs support context awareness and are emerging as the first viable commercial implementation of MANETs (mobile ad-hoc networks) (Lin et al., 2008; Raya et al., 2006)

VANET is a relatively new technology that enables vehicular communication. A number of companies have managed to introduce products that enable vehicle Internet access. An example of this is the TracNet system, by Microsoft and KVH Industries, which turns the vehicle to a Wi-Fi hotspot with connection to the Internet. The interest by the automotive manufacturers in the technology has gathered momentum in recent years to the point where new standards called the IEEE 1609 WAVE (wireless access in vehicular environment) have started to emerge. The standards basically include enhancements to the IEEE 802.11 in order to support wireless communication among vehicles as well as road side units (Lin et al., 2008; Jiang et al., 2006). However, most of the work done until recently has tended to concentrate on the development of an appropriate MAC (medium access control) layer as well as applications and services.

VANETs are expected to offer tremendous benefits. However, such networks have a number of novel problems that need to be resolved before they get implemented in a practical setting and people have the confidence to use them. Most of the problems are

associated with the security and privacy of VANETs. The major challenges to solve these problems are due to the infrastructureless and high dynamic nature of VANETs. A lot of effort has been put recently to resolve these issues in an efficient and robust manner.

This chapter discusses the security and privacy challenges associated with intelligent VANETs, along with some possible solutions. Following this introduction, Section 2 presents the characteristics of VANETs. Next, VANET security threats and challenges are described. Then Section 4 discusses possible VANET security schemes and their underlying concepts. Next, Section 5 describes an efficient light weight identity based cryptosystem (IDBCS) for VANETs. Furthermore, Section 6 describes a complete system that implements the proposed IDBCS. Finally, Section 7 presents the conclusions of this chapter.

2. VANETs Characteristics

A pervasive (or ubiquitous) network (PN) is a term that refers to a relatively newly emerging technology. It signifies the ability of users to obtain the services and applications of several distinct networks regardless of their location or time. In other words, users can choose to communicate with **Anyone**, **Any organization**, **Anytime**, **Anywhere** through **Any** network using **Any** type of device (A6), if such networks are deployed (Yeun et al., 2005; Theng & Duh, 2008). Figure 1 illustrates the basic concept of pervasive networks.

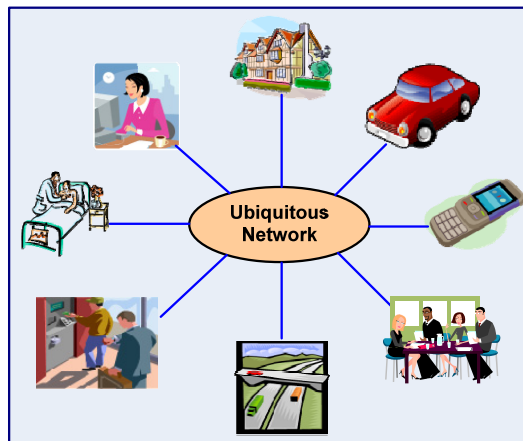


Fig. 1. Pervasive networks

This would provide users with many advantages, such as the ease of communication and the aptitude of linking diverse services into single access points. Due to such desirable features; a lot of effort is being conducted to provide the services for pervasive networks. However, a key issue that needs to be considered is the security of these networks. Currently, there are many security challenges facing PN; how secure such networks are and what are the best methods of delivering such services still remain open problems that need to be addressed (Yeun et al., 2005; Want & Pering, 2005; Connelly et al., 2008).

There are many forms of pervasive networks; most commonly the so-called Wireless Ad-hoc Networks (WANETs). As such networks are based on wireless communication, they provide ease of access but in many instances they are considered less secure than other

communication systems. ‘Ad hoc’ means that, in such networks, users or ‘nodes’ are constantly communicating with each other. In other words, these networks are based on node-to-node communication. A node can either be a user who desires certain features or dedicated equipment to manage the service.

Based on the fundamental concepts of WANETs, many other categories have emerged. The most common are: wireless mesh networks, wireless sensor networks and Mobile Ad-hoc Networks (MANETs). The former two categories proved to be useful and are used in some fields like mobile devices’ communication and weather monitoring, respectively. Whereas MANETs are those networks that offer high levels of mobility for users and take many forms. One of the most useful forms of MANETs is VANETs, which are also considered the first commercial application of MANETs (Yu & Chong, 2005; Kiess & Mauve, 2007). Figure 2 shows the general breakdown of wireless ad-hoc networks.

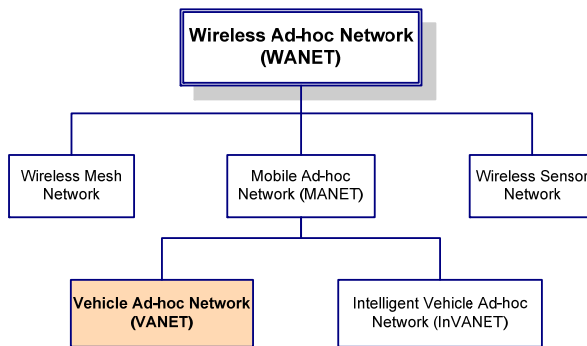


Fig. 2. Hierarchy of wireless ad-hoc networks

VANETs are wireless ad-hoc networks where the nodes, be it vehicles or RSU, can communicate and exchange data for purposes of information inquiry or distribution. The ultimate goal of VANETs is to enhance the driving experience and increase the level of safety for drivers. This can be achieved by allowing nodes within certain ranges (typically 5-10 Km) to connect with each other in order to exchange information (Raya et al., 2006; Nadeem et al. 2005; Dornbush & Joshi, 2007; Schoch, 2008). Figure 3 shows a general view of VANETs structure.

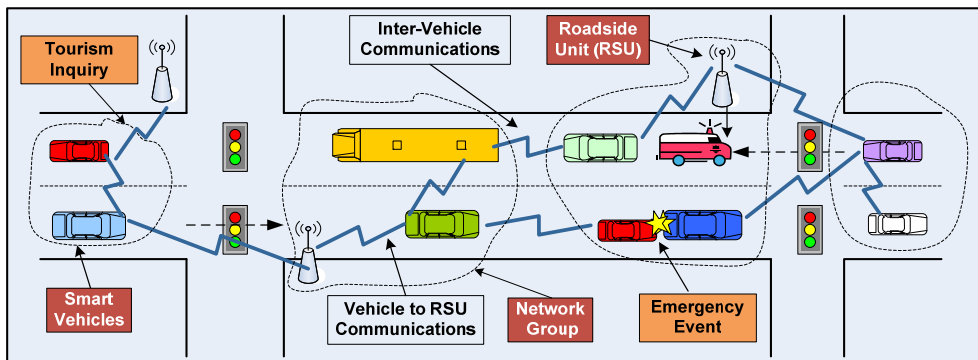


Fig. 3. The basic structure of VANETs

VANETs have a number of distinctive properties that need to be taken into consideration when designing systems to secure them. Those properties include:

- **The nature of communication:** VANETs are based on node-to-node communication; where nodes establish connections with other nodes in order to exchange information of different nature. This topology is referred to as a 'distributed' network; however in some cases, VANETs can be of a 'centralized' nature where a single authority has higher level of control. Moreover, nodes can rely on other nodes to make decisions about route selections for example. Communication can take many forms, for example a node can specifically request some information from another node, or RSU can exchange information with nodes as they pass by for database updates and so on. Obviously, this nature of communication raises many security issues which will be discussed later. Furthermore, a node in VANETs can either act as a host requesting data or a router distributing data.
- **Mobility & Dynamic-nature:** since VANETs are one form of MANETs, the 'mobility' feature is expected to be inherited. In VANETs, nodes are constantly changing their locations (except RSUs) with different speeds and directions, which make the network very dynamic in nature. For instance, a number of nodes can communicate once a group is set up. But the group can rapidly change its structure if a node leaves the group or another join it, as shown in Figure 4. This, in turn, makes it challenging to establish security protocols for a group of nodes or even to guarantee that communication is successful.
- **Frequent exchange of information:** because nodes are very mobile in VANETs, it is expected that nodes are continuously exchanging information with any number of other nodes.
- **Real time processing & self-organizing:** because of the properties mentioned above, VANET communication requires fast processing of information that does not take time in order to correctly exchange information. Furthermore, since nodes are mobile, the network is organized in different 'topologies' each time.

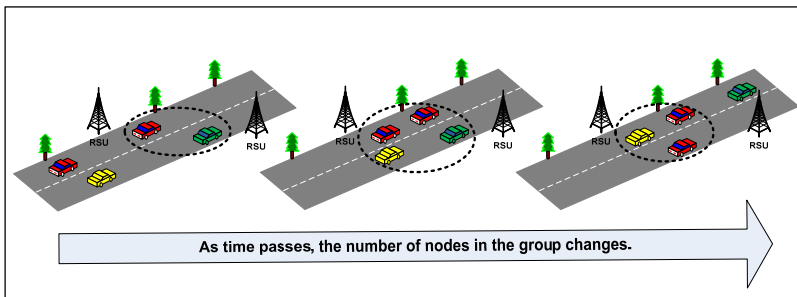


Fig. 4. The dynamic nature of VANETs

- **Infrastructure-less nature:** nodes are not connected by any sort of physical mediums in VANETs; it is completely based on a wireless 'infrastructure-less' environment. This, as will be discussed later, raises some security concerns in managing VANETs communication.

- **Low volatility:** obviously, in VANETs nodes are only in range of communication for short period of times. This causes context and data to be changed rapidly because there are many data being exchanged as nodes travel.
- **Data value vs. distance:** VANETs can provide communication over 5-10 Km ranges. As mentioned previously, nodes form virtual groups to connect and exchange information. However, the value of the information decreases as data travels further from the origin. This is because information is susceptible to various types of attacks which affect its validity (Golle et al., 2004; Zhao & Cao, 2008).
- **Other properties:** there are other properties that concern the physical and statistical aspects such as: no two nodes may exist in the same location at the same given time or that nodes rarely travel at an average speed greater than 120 Km/h. These properties help in producing more rigid security protocols.

VANETs can provide many applications that are safety or entertainment oriented. Examples include: the provision of road conditions information, traffic conditions, accident reporting which help the authorities to maintain road status, entertainment and internet access and many more (Raya et al., 2006; Boukerche et al., 2008). The diversity of applications is driven by the fact that VANETs are ultimately considered a form of pervasive networks.

Figure 5 illustrates some of these applications as they could occur in VANETs. In scenario 1 after the accident occurs; the vehicles involved in the accident notify RSU 1, which notifies RSU 2 and RSU 3 about the accident. RSU 3 notifies authorities about the location and recommends alternative routes for vehicles headed towards that location. In scenario 2 a new vehicle to the town can communicate with a RSU to provide it with directions to a nearby gas station.

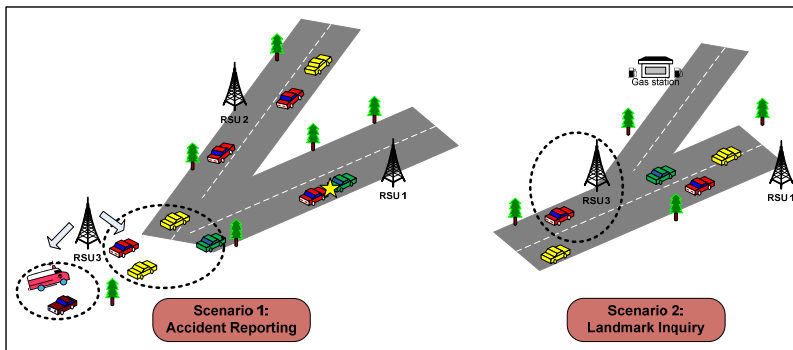


Fig. 5. Some applications of VANETs

Currently, communication in such networks as VANETs is based on the IEEE 802.11 (i.e. Wi-Fi) standards with its different enhancements (802.11b/g). Some applications of VANETs; such as toll payments, also rely on DSRC standard. However, these methods introduce some latency problems which are intolerable in such networks. Therefore, the IEEE is developing a new enhancement to the 802.11 standard that will improve communication for such network. The new standard, known as IEEE 802.11p, will be based on DSRC but with an addition of WAVE. This will support both Vehicle-to-Vehicle (V2V) and Vehicle-to-RSU (V2R) communication in VANETs (Eichler, 2007; Yang & Wang, 2007).

3. VANET Security Threats and Challenges

This section discusses the types of security attacks and network adversaries that can pose a threat for VANETs. It also highlights the major security and privacy challenges facing them.

3.1 Adversaries

Just like any other wireless network, there are many different catastrophic attacks that can occur in a VANET. Before we classify these attacks, it is good to look at how adversaries are categorized. A node is considered 'adversary' if it attempts to inject any type of misbehavior in the network that might cause other nodes (i.e. victims), and ultimately the network, to function improperly (Haubaux et al., 2004; Papadimitratos et al., 2008). Attackers can be of several forms each with different levels of impact on the network, some of these are:

- **Drivers looking only for their best interest:** for example a node might deceive other nodes that a certain route is blocked in order to clear the path to its (adversary's) destination.
- **Users who misuse VANETs:** for example a robber might try to extract data from the network to help him locate places with no cars (i.e. most likely no people), and hence break-in to that place which could be a house.
- **People from within the industry:** this category has a major impact on the security of VANETs since they can access core information about a vehicle (node) as it is manufactured.
- **Malicious attackers:** this is considered the most dangerous category since they can cause severe damage to the network. Possible attacks from this category can be ranged from eavesdropping to terrorism attacks.

3.2 Attacks and Threats

As in other communication networks, there are numerous attacks that can disturb the security of the VANET and the privacy of its nodes. Each type of attack affects some of the security services in the system; termed 'CIA' which stands for Confidentiality, Integrity, and Accountability, and Availability. In general, attacks fall into 4 categories (Maiwald, 2003) as shown in Table 1.

Type of Attack	Definition	Affected Characteristic
Access	An attempt to obtain unauthorized information.	Confidentiality; because information is exposed to unauthorized parties.
Modification	An attempt to alter and change information that is unauthorized to change.	Integrity; because correctness of information is compromised.
Denial of Service	An attempt to deny usage or access of information for legitimate users.	Availability; because the services of a network might not be available for users
Repudiation	An attempt to give incorrect information or deny the occurrence of events.	Accountability; because information is no longer liable.

Table 1. Categories of Attacks

Below is a listing of the most common and devastating forms of attacks that a VANET can suffer:

- **Denial of Service (DoS):** a very simple, but yet lethal attack. In this attack a node might continuously send unwanted data across the network so that it enters a grid-lock state where other nodes are unable to communicate due to channel blocking. This attack can either deny access to information or applications or even the whole VANET (Maiwald, 2003; Parno & Perrig, 2005; Raya & Hubaux, 2007).
- **Interception:** where a node plays the 'man-in-the-middle' role so that information exchanged between two nodes passes by the adversary node. Hence, it gains information that is intended to other destinations.
- **Fabrication:** where attackers send incorrect information to other nodes for different purposes. For example, a node sending false data about traffic conditions in certain roads. These types of attacks can be very dangerous because they affect the validity of the data received by nodes (Parno & Perrig, 2005; Raya & Hubaux, 2007).
- **Impersonation:** attackers can pretend to be what they are not in order to gain access to certain information or to aid other attacks by pretending to be a vehicle when in fact it is a stationary adversary (Raya et al., 2006; Raya & Hubaux, 2007).
- **Alteration & suppression of data:** in these types of attacks, adversary nodes can receive valid data, alter it and resend it to other nodes. Moreover, an adversary can prevent communication between two nodes by dropping certain messages between them. These attacks cause false data and confusion to be distributed among the network nodes and hence it affects performance (Golle et al., 2004).
- **The Sybil attack:** a malicious node attempts to make other nodes, which in turn, make other nodes malicious and hence control significant portions of the network and misuse it. This attack is as dangerous as DoS attacks because it can destroy valid communication in the network (Golle et al., 2004; Yan et al., 2008).

3.3 VANET Security and Privacy Challenges

VANETs are considered a relatively new area of research. Many research papers explore the nature of VANETs and how to implement them. However, recently the network security design issues of VANETs are becoming a major area of interest for researchers in order to enable assure future users of the robustness of such system (Raya et al., 2006; Lin et al., 2008; Jakubiak & Koucheryav, 2008).

One of the major challenges of securing VANETs is *communication security*. This aims to provide secure communication between vehicles, which is referred to as Inter-Vehicle Communication (IVC), and between vehicles and Road Side Units (RSU); Vehicle-to-RSU communication (VRC). Any security framework must ensure that basic security services are provided in VANETs. These services include: information confidentiality which aims to prevent unauthorized access to information. Also, integrity of exchanged messages must be provided in order to detect and prevent malicious intent such as information alteration.

Additionally, node authentication is important to ensure that all nodes within the network are who they claim to be and hence prevent impersonation. Other services include: availability of network services for all users at all times and accountability which aims to associate events with particular nodes for future references in order to prevent attempts to provide false claims or reject true ones (i.e. a node claiming that it was not at a certain location; where in fact it was) (Raya et al., 2006; Maiwald, 2003). A lot of work has been done

to achieve security in VANETs; the use of cryptography primitives such as encryption and digital signatures proved to be able to provide security services discussed above (i.e. confidentiality, integrity, authentication, etc.) in vehicular networks.

Another salient challenge that faces the security of VANETs is *key management*. The key in the security domain is the number sequence that is used to encrypt and decrypt information. The issue of key management has many categories that must be resolved when designing security protocols for such networks. One category is key revocation which is the process of discarding suspected key or keys that are bound to malicious nodes. Traditional methods of revocation such as Certificate Revocation Lists (CRLs) are not suitable for VANETs due to the large scale of the network (Lin et al., 2008). A second category of this challenge is group key management since VANETs inherit the characteristic of mobility from MANETs.

Furthermore, *detection of malicious nodes and intentions* is considered the most challenging issue in VANETs so far. The reason for that is because it is easy to access data in the network and hence data validity is compromised. Consequently, it becomes much more difficult to distinguish valid data from malicious data. What makes this even worse is that in VANETs there are no guarantees that previously honest nodes do not turn to malicious nodes in the future. Furthermore, in such networks it became desired to prevent the attack before it occurs which really calls for strong security algorithms (Yan et al., 2008; Li & Joshi, 2009).

Location verification is another challenge for VANET security. Currently in VANETs, position coordinates can be verified using either a GPS unit, a RSU, or via inter-vehicle communication (IVC). All of these methods are considered weak since an attacker can easily fool a GPS unit or manipulate RSUs or even forge data via IVC. Position verification plays a vital role to prevent many attacks like impersonation. It also helps in the data validation process. Therefore, a solid method to verify nodes positions' is required to help improving the security of VANETs (Haubaux, 2005; Golle et al., 2004; Yan et al., 2008).

These two challenges are quite significant because they intervene with the privacy of the node, i.e. drivers are not willing to reveal their routes and driving habits to be exposed by others. Consequently, they lead to another major challenge in securing VANETs which is *privacy preservation* (Rahman & Hengartner, 2007; Raya & Hubaux, 2007; Wang et al., 2008).

The privacy issue is concerned with protecting personal information of drivers (name, location, plate number, etc.) within the network. The network protocol has to be designed in a way that hides this information from other nodes; but allows it to be extracted by authorities in cases of accidents or malicious intent as a mean of auditing for authority usage. Hence, achieving 'conditional' privacy is desirable for VANETs rather than unconditional privacy which is a major challenge. Moreover, the tradeoff between robustness measures, such as the inclusion of personal information during communication which makes the task of malicious node detection easier, and the protection of drivers' information makes the issues of privacy more challenging (Lin et al., 2008; Yan et al., 2008).

The *trade-off between robustness and the level of privacy* a protocol grants is also a key challenge facing VANETs. Any proposed security algorithm must take into consideration the impact on the users and how well will the public accept it because their privacy is involved in such matters. This becomes a problem when an algorithm mainly depends on personal data as unique identifiers, in order to be robust enough, that can be traced back to a specific user. For example, the public might consider it intrusive if the algorithm requires the use and exposure of their biometric data. Hence, proposing a security protocol that is robust enough

to secure VANETs communication, yet be well-accepted by the public is still an open problem (Raya & Hubaux, 2007).

Other challenges facing VANETs include *time sensitivity* and *network scale*. The time required to process information in such networks is vital because as mentioned previously, nodes are only within the communication range for short period of time. This forces communication methods to be of real-time processing nature because nodes need to exchange, verify and prevent attacks as they are travelling at high speeds. So, we need security methods that take this issue into consideration. It is also clear how the issue of network scale can turn to a challenge when talking about such dense networks as VANETs. Huge number of vehicle, of different origins and manufactures, makes it really difficult to manage communication and security in the network (Raya et al., 2006).

The eventual goal of VANET security protocols is to provide a vehicular communication network that is able to resist malicious activities and attacks and provide the highest possible level of node privacy. This is very challenging due to some of the unique features of VANETs such as the high mobility and the large network scale (i.e. millions of vehicles). Such features make it more difficult to design protocols that will provide secure communication and prevent many types of security attacks, as well as protect all personal information of drivers unless it is absolutely required.

4. VANET Security Schemes & Concepts

This section presents a literature review for the security of VANETs and classifies the approaches used to overcome security challenges. The section also includes an explanation of the Identity Based Cryptography as it is the center of the system developed in the chapter. Then it explains in details important cryptographic concepts that are related to this chapter.

4.1 Symmetric Key Approaches

Symmetric Key systems were the first type of cryptosystems used to secure information. In these systems, nodes can only communicate after sharing and agreeing on a secret key that is used to process communication messages. As stated previously, VANETs are a relatively new research area and the security for such networks is only starting to be a major research topic. Hence, there are not many papers that propose the use of such systems for VANET security as the attention is more directed towards Public Key and Identity Based systems. Nevertheless, this section discusses existing proposals of using Symmetric Key systems for VANET security.

In (Burmester & Chrissikopoulos, 2008) a hybrid system that uses both Symmetric and Public Key operations is proposed to provide security for VANETs. The hybrid system provides authentication, confidentiality and privacy preservation. To achieve this it defines two types of communication within VANETs: pair-wise and group communication. The former type occurs when two nodes require exchanging messages, whereas the latter is established when more than two nodes require communication. They propose the use of symmetric keys when pair-wise communication occurs in order to avoid introducing overhead of using a key pair (i.e. public key systems). However, they point out that symmetric keys should not be used in the authentication process since it might prevent non-repudiation. The symmetric key generation process is explained in (Burmester & Chrissikopoulos, 2008) and the key size is 1024 bits and they suggest the use of the

Advanced Encryption Standard (AES) (a symmetric key scheme) for the encryption process (Daemen & Rijmen, 2002; Stallings, 2002).

4.2 Public Key Approaches

Public key schemes were most widely used prior to the introduction of ID-based ones. In Public Key frameworks, each node is granted a pair of keys: a secret key and a public key. These are used in security operations when communicating with other nodes. It is very important to note that in order to implement this framework; a Public Key Infrastructure (PKI) is required to handle key management operations. Based on such frameworks, the security protocol can also offer desirable features such as certificate revocation and privacy of nodes. Related works in these two fields are discussed in this section.

In (Hubaux et al., 2004) security and privacy issues in vehicular communication are addressed. They highlighted how privacy concerns arose due to the fact that the license plates were replaced with electronic identities as a method of tracking vehicles used by authorities. They proposed the use of public key cryptography in vehicular communication in order to allow authorities and vehicles to certify identities of other vehicles; using 'Electronic License Plates' (ELP).

They also suggest desirable privacy protocols that preserve drivers' personal information and mention some applications that could use the ELP. Solutions are also proposed for some types of attacks like impersonation. To ensure privacy preservation, they point out that privacy protocols must be based on anonymity schemes that hide the relationship between drivers' information and some random identifier. The article also addresses the problem of location verification in vehicular networks. It argues that GPS-based systems have more weaknesses than strengths and hence proposed the use of distance bounding protocols for the purpose of location verification in vehicular networks.

In (Raya et al., 2006), another new architecture is proposed where vehicles have two extra hardware units; the Event Data Recorder (EDR) to record all events and the Tamper-Proof Hardware (TPH) that is capable of performing cryptographic processing. The article argues that the proposed architecture provides authentication, authorization and accountability. They suggest the use of public key cryptography with a manageable and robust PKI since symmetric key cryptography do not support accountability. Authentication is performed by digital signatures of communicated messages; they proposed the use of Elliptic Curve Cryptography (EEC) since it reduces the processing requirements.

4.2.1 Certificate Revocation

In (Raya et al., 2006) a security architecture for vehicular communication that aims to provide security services for such networks is proposed. They highlight the salient challenges facing vehicular networks such as: the network scale, the privacy issues and the real-time requirements. They also describe the types of security threats and attacks that such network are susceptible to such as: impersonation, information forgery and tempering with traffic. They also proposed a novel certificate revocation technique through three protocols: the Revocation protocol of Tamper-Proof Device (RTPD), Distributed Revocation Protocol (DRP) and Revocation protocol using Compressed Certificate Revocation Lists (RCCRL). These protocols are introduced since they argue that standard methods of revocation such

as Certificate Revocation Lists (CRLs) causes substantial amount of overhead and requires pervasive infrastructure.

Furthermore, (Lin et al., 2008) discussed the current standards for providing security in vehicular communication. They described how the IEEE 1609 WAVE standards (i.e. DSRC) supports security for IVC and VRS. The IEEE 1609.2 standard provides security measures that require the use of public key cryptography with ECC support for some applications. However, drivers' privacy preservation issues are not addressed in these standards. The Vehicle Safety Communication (VSC) project by the US Department of Transportation resolves the privacy issues through the use of CRL.

The articles explain the disadvantages that prevent such methods of being suitable for vehicular environments; such as the network scale which is substantial in VANETs and causes the CRL to grow rapidly and hence increase processing requirements when revocation is required. Furthermore, it is highlighted the CRL are considered centralized approaches which do not suit VANETs because of the property of high mobility.

A novel certificate revocation scheme termed RSU-aided Certificate Revocation (RCR) is proposed by (Lin et al., 2008). In this method, the TTP grants secret keys for each RSU which enables it to sign all messages communicated within its range. Whenever a certificate is detected to be invalid; the CA issues a warning message to all RSUs which in turn use broadcast messages to all vehicles in respective ranges in order to revoke the particular certificate and stop all communication with that node. They also explain silent attacks (i.e. where a node disables message broadcasting feature in order to be camouflaged from the RSU). Figure 6 is adopted from (Lin et al., 2008) and illustrates the novel RCR method.

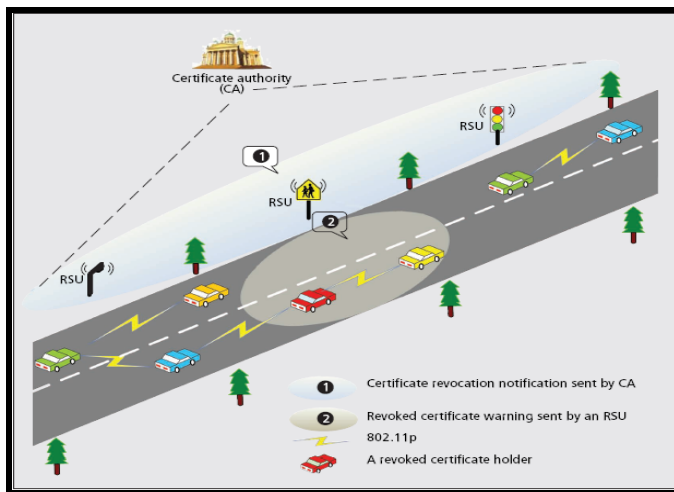


Fig. 6. RSU-aided certificate revocation

4.2.2 Pseudonym Based Approaches for Privacy

In (Raya, et al., 2006) a novel approach for privacy preservation is proposed by using of a set of anonymous keys, which have short life-times, that is previously stored in the TPD for a certain amount of time, i.e. a year or several months. Once a key is used it is declared void and cannot be used again and all key distribution and management is performed by the CA

of the network. However, they stress on the point that these keys have to be traceable to the driver only in case of emergencies or authority requirements.

In (Lin et al., 2008) the 'conditional' privacy preservation in VANETs is addressed. This is a desirable characteristic for VANET because it ensures that recipients are not able to extract senders' personal information; however, authorities are able to do so in cases of accidents or network misuse. They explain why the pseudonym-based approaches are not suitable for VANETs since at each revocation process, the CA is required to search exhaustively a large database. Moreover, as the network scale grows larger, CRL become very difficult to manage. They explain the previously proposed scheme for conditional privacy in (Lin et al., 2007); the Group Signature and Identity-based Signature (GSIS).

The scheme categorizes the process into two groups: On Board Units (OBS) to OBU and RSU to OBU; which ultimately refers to IVC and RVC. The first group uses short-group signature schemes to ensure the anonymity of communicating nodes, and IBS are used in the second group where all RSU messages are signed and the overall cryptographic overhead is reduced since it is an identity-based approach. GSIS also prevents what's called the 'RSU replication attack where a compromised RSU is relocated in order to misuse the network and spread malicious data.

4.3 Identity-Based Cryptography

Recently, this approach became the mainstream for VANET security frameworks as it is considered a viable choice due to the properties of VANETs. As mentioned previously, earlier proposed security schemes relied on the use of public key cryptography (PKC) and/or symmetric key cryptography (SKC). However, recent researches discovered that such cryptography methods are not the 'best' choice for security in VANETs. One important characteristic of VANETs is that they are of infrastructure-less nature; hence the use of PKC is not suitable since it requires a Public Key Infrastructure (PKI) which deals with issues of key distribution and management. Moreover, sizes of the keys and certificates pose a constraint on the use of PKC in such networks since the bandwidth is limited in such dynamic wireless environments. Also, because VANETs require real-time responses and cannot tolerate delays in communication; SKC is also not considered a good choice (Kamat et al., 2006). Therefore, IDBC is currently considered a viable choice to provide security in VANETs.

4.3.1 Identity-Based Signature

The basic idea of identity-based signature (Shamir, 1984) is to provide secure communication without the requirement of a public/private key pair. IDBC is based on an underlying public key cryptosystem. However, instead of generating a key pair, an arbitrary string that uniquely identifies the user can be used as his public key. The private key is then generated by a Third Trusted Party (TTP) and issued to the user (Shamir, 1984). However, (Shamir, 1984) was only able to propose a functional Identity-Based Signature (IBS) scheme but not an encryption scheme.

As stated previously; IDBC requires an underlying public key cryptosystem, but Identity-Based Encryption (IBE) scheme requires two additional requirements: the ability of easily computing private keys from a random seed and the intractability of the process of computing this seed if a public/private key pair is known. At that time, the proposal used

RSA (Shamir, 1984) as the underlying public key cryptosystem which did not satisfy the two additional requirements for an IBE scheme, and hence it was an open problem. Figure 7 below illustrates a general view of the proposed IBS scheme by (Shamir, 1984).

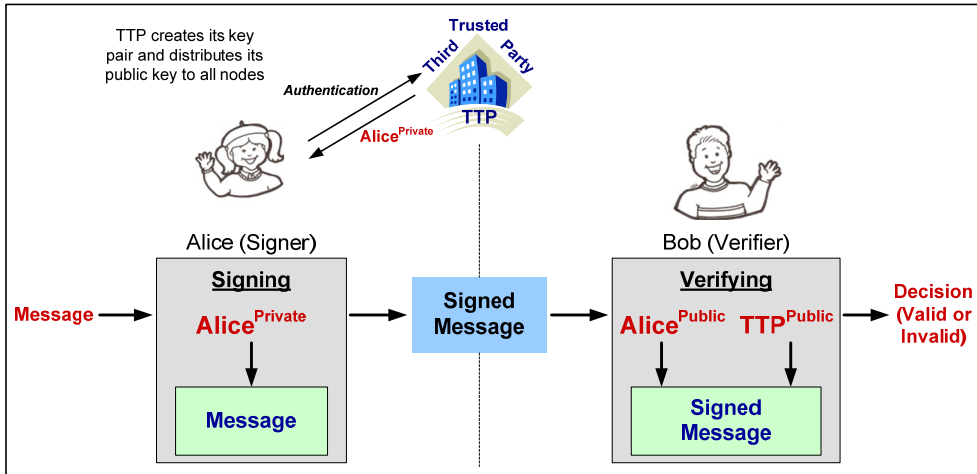


Fig. 7. A General Identity-Based Signature Scheme

The signing/verifying process is performed in 4 steps (Baek et al., 2004):

- **Setup:** TTP creates its own pair of public and private key (master secret) and distributes its public key to all parties within the network.
- **Extraction:** the signer of the message (Alice) authenticates herself to the TTP and requests her private key ($Alice^{pri}$), which is generated by the TTP and issued to Alice.
- **Signing:** the signer (Alice) uses her private key ($Alice^{pri}$) to sign the message and send it to Bob.
- **Verifying:** Upon receiving the signed message, the verifier (Bob) uses the public key of Alice ($Alice^{pub}$) and the public key of the TTP (TTP^{pub}) to make a decision whether the signature is valid or invalid.

4.3.2 Identity-Based Encryption

The IBE open problem was solved by (Boneh & Franklin, 2001) with a fully functional scheme based on the Weil Pairing (Stinson, 2005). The strength of the scheme they proposed was based on the intractability of the Elliptic Curve Discrete Logarithm Problem (ECDLP), which will be discussed in the later.

The encryption process shown in Figure 8 is performed in 4 steps (Baek et al., 2004):

- **Setup:** TTP creates its own pair of public and private key (master secret) and distributes its public key to all parties within the network.
- **Extraction:** the recipient (Bob) authenticates himself to the TTP and requests his private key (Bob^{pri}), which is generated by the TTP and issued to Bob.

- **Encryption:** the sender (Alice) uses the Bob's public key which is the arbitrary string (Bob^{ID}) and the public key of the TTP (TTP^{pub}) to encrypt the message and send it to Bob.
- **Decryption:** Upon receiving the encrypted message, Bob uses his private key (Bob^{pri}) to obtain the original message.

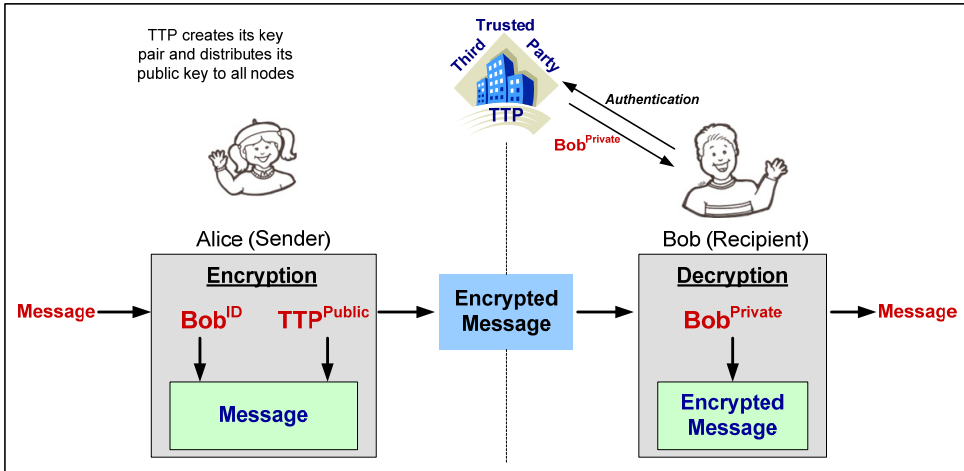


Fig. 8. A General Identity-Based Encryption Scheme

The strength of the security offered by IDBC is based on four key points as stated in (Shamir, 1984):

1. The strength of the underlying public key cryptosystem.
2. The level of secrecy of all information acquired and stored in the TTP.
3. The strength of the authentication methods performed prior to private key issuance.
4. The methods and precautions by which private keys are guaranteed not to be leaked.

4.3.4 Identity-Based Approaches

Few researchers proposed the use of IDBC for VANET security. In (Sun et al., 2007) an ID-based framework is presented that could achieve privacy and non-repudiation; along with the fundamental security features, in VANETs. The importance of having privacy preserved in such network is highlighted as a key issue to attract vehicles to join such vehicular networks. The proposed framework includes a justification as to why previously proposed ID-based solutions to achieve privacy; such as ring signatures, do not suit VANET environments since it results in 'unconditional privacy'. The latter term refers to the inability to reveal the identity of vehicles under all circumstances; which should not be the case in VANETs. This is resolved by (Sun et al., 2007) through the use of 'distributed control', where a single authority is unable to reveal drivers' personal information. Instead, multiple authorities can participate in a collaborative process in case an identity needs to be revealed for legal reasons.

The framework relies on the pseudonym-based approach to achieve non-repudiation in VANETs. This approach was introduced previously in (Raya et al., 2006) and it involves

preloading vehicles with a set of short-lived keys that cannot be used more than one time, hence other vehicles are unable to track the identity of particular vehicles. They proposed the addition of a Pseudonym Lookup Table (PLT) that can be used to associate random identifiers (pseudonyms) with the real identity of the vehicle.

In (Sun et al., 2007) the authors also suggest the use of existing wireless infrastructure to perform key revocation processes since there does not exist a dedicated vehicular communication infrastructure. However, the proposed framework assumes the use of Tamper-proof Hardware (TPH) which ensures that the master secret of the TTP is never disclosed. Although the proposed framework is based on IDBC; they also require the use of public or symmetric key cryptography for further communication once mutual authentication has been established between nodes in VANETs. The proposal suggests a method based on ID-based threshold signatures to provide non-repudiation services for authorities in VANETs.

In (Kamat et al., 2006) another IDBC is proposed for VANET security. It stressed on the indispensability of security and privacy in VANETs in order for them to be well-accepted by the public. They point out that VANET nodes should be able to mutually authenticate with other nodes; but protect the identity of themselves in order to grant privacy services. It explains why traditional cryptography techniques cannot be used in VANETs environments and why IDBC is possibly the 'best' solution to resolve VANET security issues.

The paper also proposes the use of 'signcryption' (Baek et al., 2007) which have considerable advantages over standard encryption and signature methods. Such advantages include reduced cipher sizes but they argue that VANET nodes are not computational-power restricted and can perform complex processes of Tate pairings. Their proposal suggests that the base station is the only party that will require storing CRLs, hence preserving a substantial amount of storage space in vehicles. Moreover, the issue of non-repudiation is also addressed and explained in details in (Kamat et al., 2006).

4.4 Cryptographic Mathematical Concepts

The field of cryptography is very much based on the number theory and other mathematical concepts (Stinson, 2006; Washington, 2007; Menezes et al., 1997). There are several mathematical terms that are used in cryptography; this section explains the most important ones. Most of the notations used in this section will be reused later in this chapter.

- **Roots of Unity:** all complex number that yield the value of 1 when raised to a given power n . Also referred to as de Moivre numbers (Conway & Guy, 1995); they can be represented on the unit circle of the complex plane. Mathematically, n^{th} root of unity is defined as a complex number that satisfies:

$$Z^n = 1; \quad n = 1, 2, 3, \dots$$

- **Cyclic Groups:** a group G of elements is called 'cyclic' if a generator 'g' element exist such that all the elements of the group can be represented as a power of g (the multiplicative representation) or a multiple of g (the additive representation) (Joseph, 1998). It is defined as:

$$G = \langle g \rangle = \{g^n \mid n : \text{int eger}\}$$

For example; suppose that $G = \{g^0, g^1, g^2, g^3, g^4\}$ is a cyclic, then $g^5 = g^0, g^6 = g^1$ and so on.

- **Group Generator:** a subset **S** of a group **G** is referred to as the ‘generating set of **G**’ if every element in the group **G** can be expressed as a product of a finite number of elements in the subset **S** (Arfken & Weber, 2005). If $G = \langle S \rangle$ then it is said that **S** generates **G** and the elements of **S** are called generators of **G**.
- **Group Order:** the order of a group **G** is defined mathematically as the number of elements in the group (i.e. the group’s cardinality) (Arfken & Wber, 2005). For example, if $G = \{1,2,4,7\}$ then the order of **G** is denoted as:

$$|G| = 4 \text{ OR } ord(G) = 4$$

- **Abelian Groups:** a group is called ‘Abelian’ if operations on elements within the group do not depend on their respective orders. These groups are characterized as commutative and associative. Moreover, an inverse element exist for each element in an Abelian group and the group posses the identity element (Finch, 2003). Generally, these groups can be represented in two ways: additive and multiplicative notation. Table 2 below defined the convention for each representation.

Convention	Operation	Identity	Powers	Inverse
Addition	$x + y$	0	Nx	$-x$
Multiplication	$x * y$	e or 1	x^n	x^{-1}

Table 2. Abelian Groups' Conventions

- **Torsion Group:** a group **G** is called ‘Torsion’ or periodic if all elements within the group have finite orders. The order of an element **x** in **G** is defined as the smallest integer ‘**n**’ such that:

$$x^n = e; \text{ e is the identity element of G}$$

If no **n** exist such that the above equation is satisfied; then the element is said to have an infinite order (Armitage & Eberlein, 2006).

- **Bilinear Maps:** are defined as mathematical functions that map the product of 2 linear elements to a third element; all within the same group (Boneh et al., 2003). For example, let **X** and **Y** be linear elements of **G**, then the bilinear map is defined as:

$$F : X * Y \rightarrow Z ; Z \text{ is a third element in G}$$

An example of a bilinear map is the multiplication (i.e. a mathematical function) of elements in the integer group **N**. For instance

$$2, 3 \in N \text{ and} \\ 2, 3 = 6 \in N$$

Hence integer multiplication is a bilinear map.

4.5 Strength of Cryptosystems

It is a well-known fact that there does not exist a security algorithm that is mathematically proven to be secure (Stinson, 2006; Menezes et al., 1997). The core of any cryptosystem is a computationally infeasible problem; which is not proven to be 'unbreakable' but assumed to be computationally hard. Hence, such problems allow the use of cryptosystems that are based on the intractability of these problems. This section explains these mathematical problems.

4.5.1 The Discrete Logarithm Problem

This is considered the base mathematical problem that allows cryptosystems to be considered secure. As long as the Discrete Logarithm Problem (DLP) is computationally infeasible and cannot be solved; cryptosystems based on these problems are considered secure (Stinson, 2006). For instance, the famous ElGamal Cryptosystem is based on the assumed hardness of the DLP. The DLP is described below:

*Given a group G , $\alpha \in G$ with order n and $\beta \in \langle \alpha \rangle$
find the unique integer a such that:
 $\alpha^a = \beta$
where a is called the discrete logarithm of β*

4.5.2 The Diffie-Hellman Problems

When the Diffie-Hellman key agreement protocol was introduced; its strength was associated with the difficulty of solving the Diffie-Hellman Problem (DHP) which is described below (Diffie & Hellman, 1976):

*Given a group *generator g and some random integers α, β
if g^α and g^β are known
find $g^{\alpha\beta}$*

As this problem became very important in the field of cryptography; several variants of the problem were introduced, namely: the Computational Diffie-Hellman Problem (CDHP) and the Decisional Diffie-Hellman Problem (DDHP) which are explained below:

- **The Computational Diffie-Hellman Problem (CDHP):**
The setting of this problem is similar to the DLP problem (Stinson, 2006).

*Given a group G , $\alpha \in G$ with order n and $\beta, \gamma \in \langle \alpha \rangle$
find the unique integer a such that:
 $\log_\alpha a \equiv \log_\alpha \beta \times \log_\alpha \gamma \pmod{n}$
or more clearly, given α^x and α^y , find g^{xy}
where x and y are integers*

- **The Decisional Diffie-Hellman Problem (DDHP):**

The setting of this problem is similar to the CDHP, and the problem is to make a decision whether it is the case that the CDHP holds or not (Stinson, 2006). Equivalently, it can be described as:

$$\begin{aligned} & \text{Given } \alpha^x, \alpha^y \text{ and } \alpha^z \\ & \text{Make a decision whether the following condition holds or not} \\ & z \equiv xy \pmod{n} \end{aligned}$$

4.6 Elliptic Curves Cryptography

Elliptic Curve Cryptography (ECC) is considered a public key approach for cryptography that is based on algebraic (Abelian) elliptic curve groups over finite fields. The ECC approach allowed many existing protocols and cryptographic schemes to use it in order to have a variant of the original protocol. For example, ECC can be used to construct the Elliptic Curve Diffie-Hellman (ECDH) key agreement scheme where elliptic curves are used to agree a shared key between two parties (Hankerson et al., 2004; Washington, 2008).

As stated previously, the strength of any cryptosystem is based on a computationally infeasible problem. In the case of ECC, this computationally difficult problem is termed the Elliptic Curve Discrete Logarithm Problem (ECDLP). The source of the problem is known as the Scalar Multiplication (SM) of Elliptic Curves. The SM problem is described below:

Suppose that P is an elliptic curve point

$$\begin{aligned} & \text{Find the result } (R) \text{ of doubling } P \text{ several times } (K \text{ time}), \text{ such that:} \\ & P \cdot K = R \end{aligned}$$

The difficulty of this problem relies on the infeasible computation of the value K where; this is referred to as the intractability of the 'scalar multiplication' of the point P . In practical cryptosystems, the value of K is very large such that it is computationally infeasible to compute its value using successive doubling operation of the elliptic curve point (i.e. $P \rightarrow 2P \rightarrow 2P + P = 3P \dots \rightarrow KP$) (Hankerson et al., 2004; Washington, 2008).

4.7 Pairing Based Cryptography

IDBC can be achieved through two main methods: *quadratic residues*; which is a variant of integer factorization and *admissible bilinear pairings* (Baek et al., 2004). The former is proved to be inefficient since it relies on bit-by-bit encryption processes which results in huge cipher-texts (Baek et al., 2004; Stinson, 2006). The scheme proposed by (Boneh & Franklin, 2001) is based on admissible bilinear pairings; which is more accepted and used in the field of IDBC due to its efficiency.

The main concept of PBC is constructing a mapping between two suitable cryptographic groups (e.g. elliptic curve groups) and then being able to reduce the complexity of a problem in one group to be simpler in the other group; hence producing sufficient cryptographic schemes (Galbraith & Paterson, 2008). For instance, the DDHP and the DLP can be easily solved using these mappings (i.e. pairings).

In mathematics, pairings refer to bilinear maps which maps elements of one group to elements of another group and satisfy three conditions: bi-linearity, non-degeneracy and efficient computability. Pairings are explained below (Stinson, 2006; Washington, 2008):

*Suppose two groups $G1$ and $G2$ (which can be additive or multiplicative)
of the same prime order q
 P and Q are generators of $G1$
We consider a mapping function $e: G1 \times G1 \rightarrow G2$ that satisfies 3 conditions:*

Bilinearity:

$$\forall P, Q \in G1 \text{ and } x, y \in \mathbb{Z}_q^* \\ e(xP, yQ) = e(P, Q)^{xy}$$

Non-degeneracy:

$$\forall P \in G1 \text{ and } P \text{ is a generator of } G1 \\ \text{Then } e(P, P) \text{ is a generator of } G2$$

Efficient Computability:

There exists an efficient algorithm to compute $e \forall P, Q \in G1$

Examples of such pairings include the Weil Pairings and the Tate Pairing (Galbraith & Paterson, 2008). In both cases; one of the two groups is an elliptic curve group and the other is algebraic group of finite field. The strength of current IDBCSs relies on a third variant of the DHP; namely the 'Bilinear Diffie-Hellman Problem' (BDHP) which is explained below (Stinson, 2006).

*Given $G, q, e, P, aP, bP,$ and cP
where a, b and c are random elements $\in \mathbb{Z}_q^*$
computing $e(P, P)^{abc}$ is assumed to be hard*

It is important to note that the cryptosystem designed and implemented in this chapter is based on bilinear pairings using the Weil pairing on elliptic curve groups (Boneh & Franklin, 2001).

5. Identity Based Cryptosystem for VANETs

This section explains the specifications of the Identity Based Cryptosystem (IDBCS) developed for VANETs security. It explains the system architecture through the functional, behavioral and data models of the system.

5.1 Functional Model

The main goal of the system developed in this chapter is to demonstrate how VANET security could be achieved using Identity Based Cryptography. The system consists of the main 4 functions of Identity Based Cryptography: setup, extraction, encryption and

decryption; in addition to other functions that are required to construct a complete cryptosystem. Figure 9 shows the *functional decomposition* of the IDBCS.

The IDBCS can be decomposed into the following main modules:

- **System Setup:** this function is responsible for initializing all the parameters that will be used in the system. Parameters refer to: Pairing Based Cryptography elements, elliptic curves and pairing functions.
- **PKG Setup:** this function generates all the key elements associated with the TTP or what is referred to as Private Key Generator (PKG) in IDBC since it is responsible for generating private keys for users. Five keys are associated with the PKG: master secret, system generator, public key, secret signature key and public verifier key. This function also creates three system record files: the medium file which holds all data communicated within the system, the map file which maps messages to random numbers and the status file which stores registered users.
- **User Parameter Extraction:** this function is responsible for generating all key elements associated with the user of the system. These keys will be used in order to complete operations within the system such as: encryption or digital signatures. Similarly, four keys are generated for each user: public, private, signature and verifying key.
- **PBC Elements Management:** the system is designed to hold all secret and/or public keys of users and the PKG in respective files. This function manages the read, write, convert, extract and update operations of all elements and files.
- **User Registration & Authentication:** in order for users to communicate messages with other users using the system; they should first go through a registration process. This function is responsible for acquiring user information, validating input data and creating specific files that will hold all the elements required for the user to use the system.
- **Message Communication:** this function performs the core functionalities of the message communication between two users. It is responsible for extracting the required parameters in order to encrypt the input message, digitally sign it in the sender's side and decrypt the message and verify the signature in the receiver's side. Moreover, it updates the files that are created for each user by these messages sent/received.
- **Check User Status:** this function simply checks if the user is registered in the system or not. If the user is not registered, it passed him to the registration process; otherwise the user is passed to the communication process.
- **System Reset:** this function flushes all PBC and system elements previously generated and deletes all record files created. Performing this function will disable all functionalities of the system unless setup is performed again.

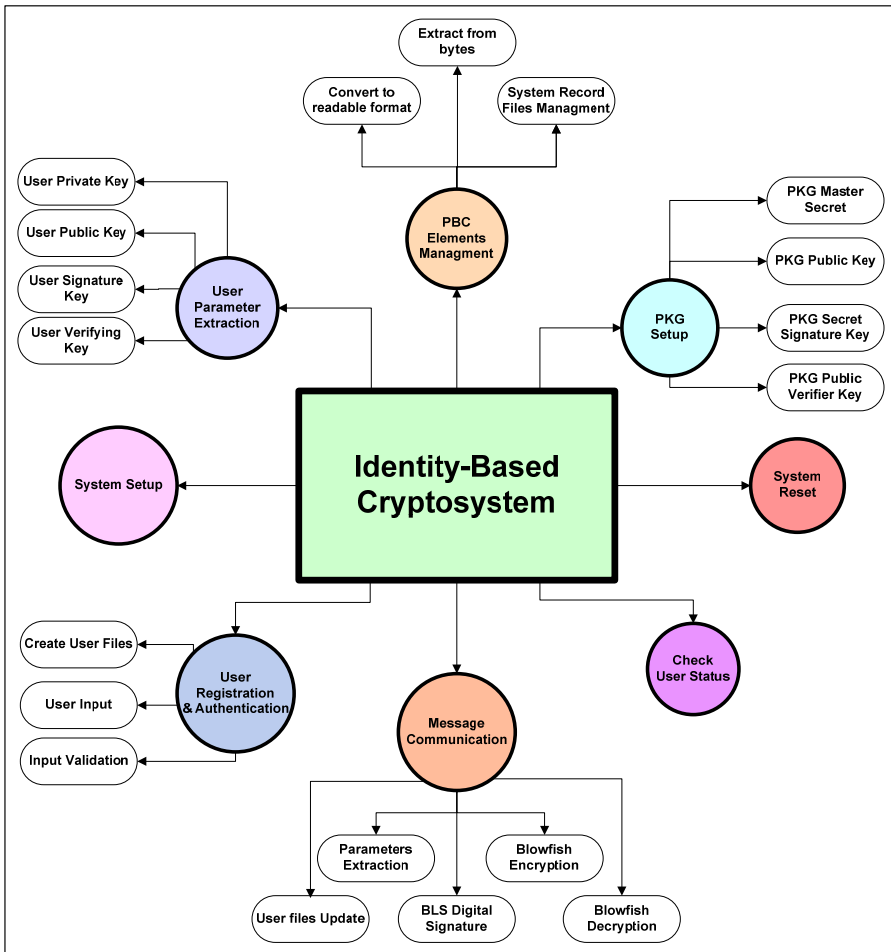


Fig. 9. The IDBCS functional decomposition

5.2 Data Model

The modules that make up IDBCS as depicted in Figure 9 communicate numerous data between them. These communication operations are critical in order for the IDBCS to operate correctly. Figure 10 shows the data flow diagram of the IDBCS.

As can be seen from Figure 10, the system requires data inputs from the user: name, date of birth, vehicle model, vehicle registration number, source and destination plate numbers and the input message the user wishes to send. Moreover, the system requires the parameters from the system setup function in order to produce correct output data for other modules. Additionally, an input is required to choose the function required to be performed.

The outputs produced by the system are directed to different modules which perform certain operation with these data. Below is an explanation of which output data are directed to which modules:

- The function choice determines which function to perform. For each function, specific messages depending on the flow of the system are displayed for the user.
- The PKG and user keys are directed to PKG and user files respectively. Some user information along with timestamp is directed to the users' status update module which adds users to the status file.
- System parameters and the encrypted and signed message are directed to the system record file update module which adds these data to respective files.
- The input message or the decrypted message, a timestamp and source/destination plate numbers are directed to the user message database module which updates the files created for the user with these data.

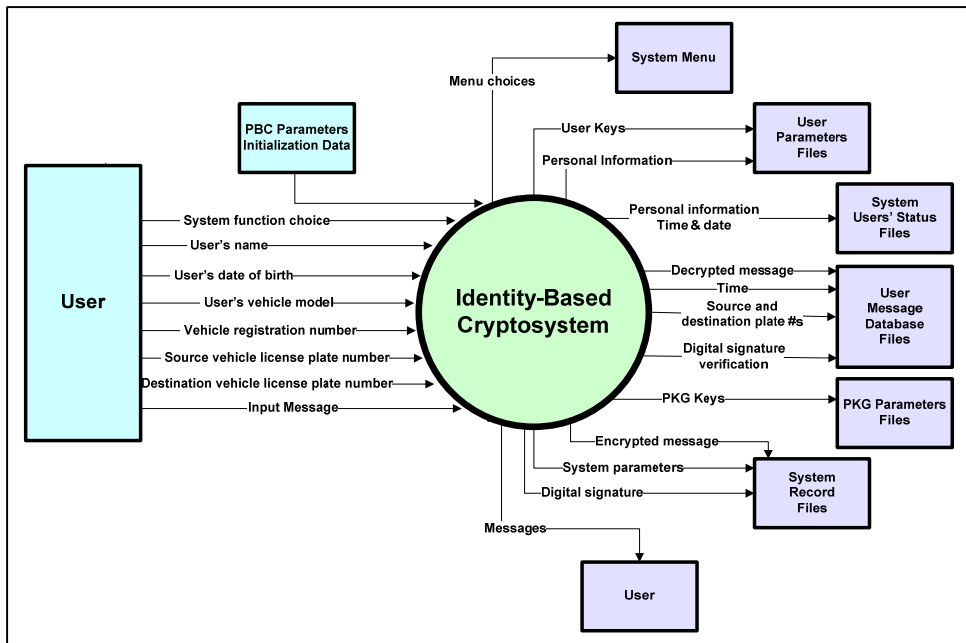


Fig. 10. The IDBCS data flow diagram

5.3 Behavioral Model

The behavior of the IDBCS is shown in Figure 11 which illustrates the State Transition Diagram of the IDBCS. As can be seen from the figure, the initialization occurs when the system is started and then it waits for choice input from the user. Depending on this choice, the system performs the corresponding function. There are 5 choices which the user could choose from:

- PKG Setup
- Registration
- Send Messages
- System Reset
- Exit.

At each choice, the corresponding function(s) is performed and then the user is directed back to the main screen for the next input (except when the choice is Exit).

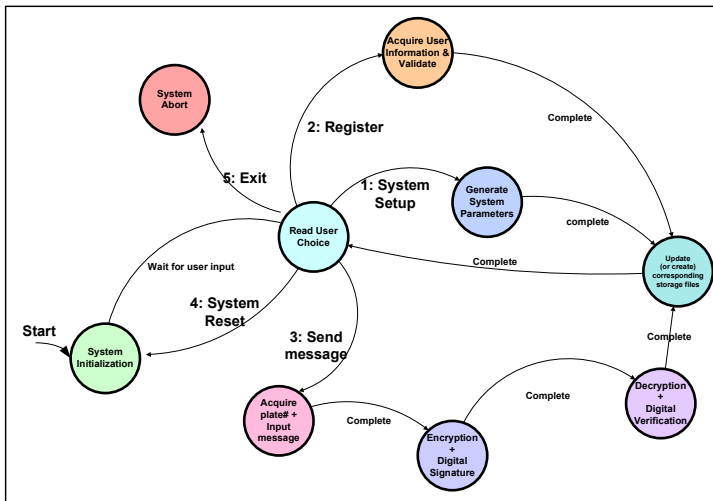


Fig. 11. The IDBCS state transition diagram

6. Complete IDBCS for VANETs

This section outlines the proposed IDBCS for VANETs as a complete system and describes its implementation. The first step in the IDBCS is the initialization of all elements and pairing functions. When that is done, the main screen is printed to the user with 5 possible functions to choose from: system setup, registration, send messages, system reset and exit. When the user chooses an option, the system calls the corresponding function, prints status messages to the user and then returns to the main screen for the next choice (except for exit). If the user chooses system setup, the system checks if there are any PKG files existing, if there are PKG files then the system only reads the keys of the PKG so that they can be used in the system. Otherwise, the setup function is called: *setup()* and when that is done a flag is set up to indicate that setup was performed successfully. The system prints a message to the user to indicate whether setup is already performed or was performed successfully now. If the user chooses registration; the system first checks if setup was performed or not by checking the flag value. If setup was not performed, then the system informs the user that registration cannot be done prior to system setup. If setup was performed, then the system asks the user to enter the car plate number and the check status function is called: *check_status(plate_num)* to check whether the user is registered or not. If the user is registered, the system informs the user that he/she is already registered and can send messages. Otherwise, the registration function is called: *registration()*, and then the extraction function is called: *extraction(plate_num, registration_num, user_file_name)*. After performing these two functions, the system informs the user that he was successfully registered to the system and can now send or receive messages. If the user chooses message communication, the system asks the user to enter his/her plate number and the check status function is called: *check_status(plate_num)* to check whether the

user is registered or not. If the user is not registered, the system informs him/her that he/she cannot communication unless he is registered. If the user is registered; he/she is asked to enter the plate number of the destination and a similar process is repeated to check if the destination vehicle is registered or not. If both vehicles are registered, message communication function is called: $msg_comm(source_plate_num, destination_plate_num)$. If the user chooses system reset, the system reset function is called: $system_reset()$. Finally, if the user chooses exit, then the system is aborted. The files created by the IDBCS can be viewed by using the command terminal: $gedit\ file_name$. Figure 12 shows the flowchart of the IDBCS.

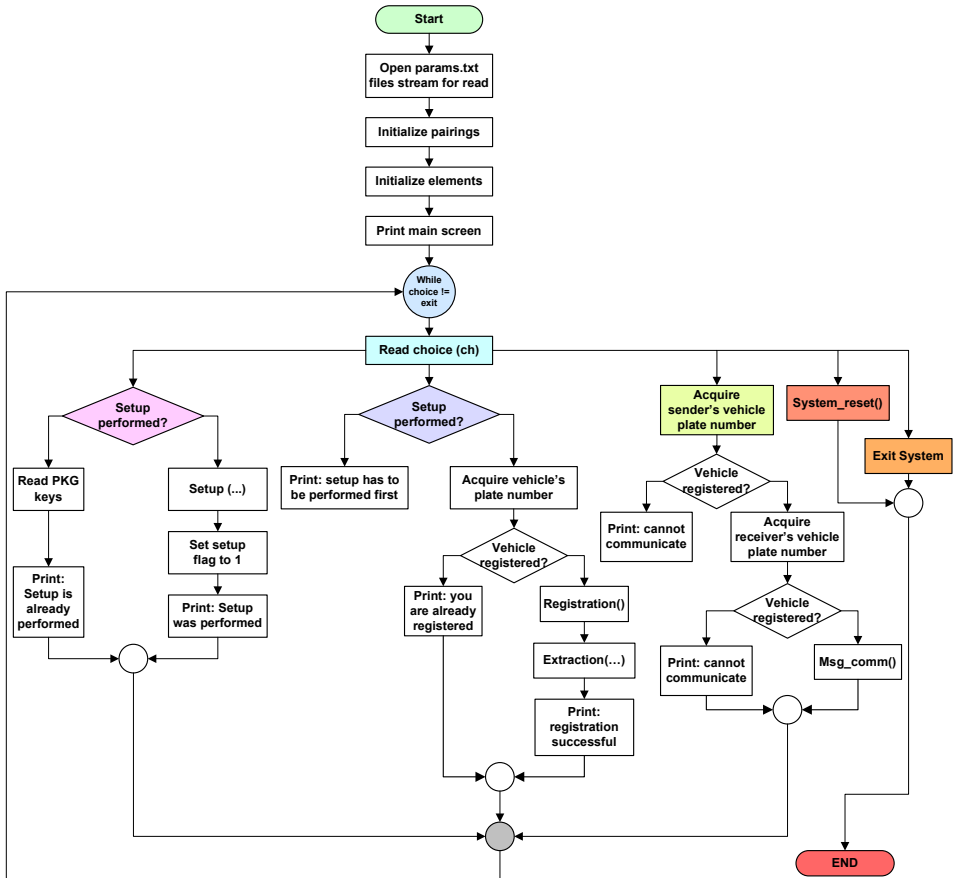


Fig. 12. Flowchart of the IDBCS for VANETs

7. Conclusion

This book chapter presented a study of VANETs. It highlighted their properties and security and privacy challenges such mobile ad hoc networks present. Furthermore, three main cryptography schemes were investigated: public key, symmetric key, and identity based

cryptography as they could be used for the security of the network. The advantages and disadvantages of these schemes were identified and overlapped with the properties of VANETs.

The study showed that identity based cryptography (IDBC) is considered the most viable choice to provide security for such networks. This is primarily due to the light-weight nature of IDBC techniques which align themselves well with the major properties of VANETs which include the infrastructure-less nature and the requirement for high speed real-time response.

In addition to the study of the various security schemes, the book chapter presented a novel implementation of an Identity Based Cryptosystem (IDBCS) that demonstrates how this scheme could be used for VANETs security. The system was designed and implemented and is based on Pairing Based Cryptography (PBC) and Elliptic Curve Cryptography (ECC). Several cryptographic primitives such as encryption and digital signature were implemented in order to provide the fundamental security services of confidentiality, integrity, authentication and non-repudiation.

Security analysis of the implemented IDBCS proved that the system is computationally secure since it implements algorithms which require a very large number of operations to break. The efficiency of the system was also measured and the results indicated that the IDBCS is computationally efficient as most of its functions do not require extensive processing or time.

8. References

- Arfken, G. & Weber, H. (2005). *Mathematical Methods For Physicists*. Academic Press
- Armitage, J. & Eberlein, W. (2006). *Elliptic Functions*. Cambridge University Press
- Baek, J.; Newmarch, J., Safavi-Naini, R. & Susilo, W. (2004). A survey of identity-based Cryptography. *Proceedings of Australian Unix Users Group Annual Conference*.
- Baek, J.; Steinfeld, R. & Zheng, Y. (2007). Formal proofs for the security of signcryption. *Journal of Cryptology*, Vol. 20, pp. 203-235
- Boneh, D. & Franklin, M. (2001). Identity-based encryption from the Weil pairing, *Proceedings of Crypto 2001*, LNCS, Vol. 2139, pp. 213-229, Springer-Verlag
- Boneh, D.; Lynn, B. & Shacham, H. (2001). Short signatures from the Weil pairings, *Proceedings of Advances in Cryptology*, LNCS, Vol. 2248, pp. 514-532, Springer-Verlag.
- Boneh, D.; Gentry, C., Lynn, B. & Shacham, H. (2003). Aggregate and Verifiably Encrypted Signatures from Bilinear Maps. *Proceedings of Advances in Cryptology - EUROCRYPT 2003*, LNCS 2656, pp. 416-432
- Boukerche, A; Oliveira, H., Nakamura, E. & Loureiro, A. (2008). Vehicular Ad Hoc Networks: A New Challenge for Localization-Based Systems. *Computer Communications*, Elsevier, Vol. 31, No. 12, pp. 2838-2849
- Burmester, M.; Magkos, E. & Chrissikopoulos, V. (2008). Strengthening privacy protection in VANETs, *Proceedings of IEEE Int. Conference on Wireless & Mobile Computing, Networking & Communication*, pp. 508-513.
- Connelly, K.; Siek, K.A., Mulder, I., Neely, S., Stevenson, G. & Kray, C. (2008). Evaluating pervasive and ubiquitous systems. *IEEE Pervasive Computing*. Vol.3, No.7, pp.85-88
- Conway, J. & Guy, R. (1995). *The book of numbers*. Springer

- Daemen, J. & Rijmen, V. (2002). The design of Rijndael: AES - The advanced encryption standard. *Springer*
- Diffie, W. & Hellman, M.E. (1976). New directions in cryptography, *IEEE Transactions on Information Theory*, IT-22, 6, pp.644-654
- Dornbush, S. & Joshi, A. (2007). StreetSmart Traffic: Discovering and Disseminating Automobile Congestion Using VANETs. *Proceedings of IEEE Vehicular Technology Conference*, pp.11-15
- Eichler, S. (2007). Performance Evaluation of the IEEE 802.11p WAVE Communication Standard. *Proceedings of IEEE Vehicular Technology Conference*, pp. 2199-2203.
- Finch, S. (2003). *Mathematical Constants*. Cambridge University Press
- Galbraith, S. & Paterson, K. (ed) (2008). Pairing-based cryptography - Pairing 2008. *Springer*
- Golle, P.; Greene, D. & Staddon, J. (2004). Detecting and correcting malicious data in VANETs, *Proceedings of First ACM Workshop on Vehicular Ad-hoc Networks*, pp. 29-37
- Hankerson, D.; Menezes, A.J. & Vanstone, S. (2004). *Guide to elliptic curve cryptography*. Springer
- Hubaux, J.; Capkun, S. & Luo, J. (2004). The security and privacy of smart vehicles. *IEEE Security & Privacy*, Vol. 2, No.3, pp. 49-55
- Kamat, P.; Baliga, A. & Trappe, W. (2006). An Identity-based security framework for VANETs, *Proceedings of 3rd Int. Workshop on Vehicular Ad-hoc Networks*, pp. 94-95.
- Jakubiak, J. & Koucheryavy, Y. (2008). State of the Art and Research Challenges for VANETs. *Proceedings of IEEE Consumer Communications and Networking Conference - CCNC 2008*, pp. 912-916
- Jiang, D.; Taliwal, V., Meier, A., Holfelder, W. & Herrtwich, R. (2006). Design of 5.9 GHz DSRC-based vehicular safety communication," *IEEE Wireless Communications*, Vol. 13, pp.36-43
- Kiess, W. & Mauve, M. (2007). A survey on real-world implementations of mobile ad-hoc networks. *Ad Hoc Networks*, Vol. 5, No. 3, pp. 423-339.
- Li, W. & Joshi, A. (2009). Outlier detection in ad hoc networks using Dempster-Shafer theory. *Proceedings of Int. Conference on Mobile Data Management, Systems, Services and Middleware - MDM 2009*.
- Lin, X.; Sun, X., Ho, P. H. & Shen, X. (2007). GSIS: A secure and privacy preserving protocol for vehicular communications. *IEEE Transactions on Vehicular Technology*, Vol. 56, No. 6, pp. 3442-3456
- Lin, X.; Lu, R., Zhang, C., Zhu, H., Ho, P. & Shen, X. (2008). Security in vehicular ad hoc networks. *IEEE Communications Magazine*, Vol. 46, No. 4, pp. 88-95
- Joseph, G. (1998). *Contemporary abstract algebra*, 4th ed.. Houghton Mifflin, USA
- Maiwald, E. (2003). *Fundamentals of Network Security*, McGraw Hill, USA
- Menezes, J. A.; Van Oorschot, P. C. & Vanstone, S. A. (1997). *Handbook of Applied Cryptography*, CRC Press
- Nadeem, T.; Shankar, P. & Iftode, L. (2006). A Comparative Study of Data Dissemination Models for VANETs. *Proceedings of Annual International Conference on Mobile and Ubiquitous Systems (MOBIQUITOUS)*, San Jose, CA, USA
- Papadimitratos, P.; Buttyan, L., Holczer, T., Schoch, E., Freudiger, J., Raya, M., Zhendong Ma, Kargl, F., Kung, A., Hubaux, J. P. (2008). Secure vehicular communication systems: design and architecture. *IEEE Communication Magazine*, Vol. 46, No. 11, pp. 100-109

- Parno, B. & Perrig, A. (2005). Challenges in securing vehicular networks, *Proceedings of the Workshop on Hot Topics in Networks (HotNets-IV)*
- Rahman, S. & Hengartner, U. (2007). Secure crash reporting in vehicular Ad hoc networks. *Proceedings of Int. Conf. Security and Privacy in Communications Networks - SecureComm 2007*, pp. 443-452.
- Raya, M. & Hubaux, J. (2007). Securing vehicular ad hoc networks. *Journal of Computer Security*, Vol. 15, pp. 39-68
- Raya, M.; Papadimitratos, P. & Hubaux, J. (2006). Securing vehicular communication. *IEEE Wireless Communication*, Vol.13, No.5, pp. 8-15
- Schoch, E.; Kargl, F., Leinmuller, T. & Weber, W. (2008). Communication Patterns in VANETs. *IEEE Communications Magazine*, Vol.46, No.11, pp.119-125.
- Shamir; (1984). Identity-based cryptosystems and signature schemes, *Proceedings of Advances in Cryptology - Crypto' 84*, LNCS, Vol. 196, Springer-Verlag, pp. 47-53.
- Stallings, W. (2002). The advanced encryption standard. *Cryptologia*, Taylor & Francis, Vol. 26, No. 3, pp. 165-188.
- Stinson, D. R. (2005). *Cryptography Theory and Practice, 3rd ed.*, Chapman & Hall/CRC, USA.
- Sun, J.; Zhang, C. & Fang, Y. (2007). An ID-based framework achieving privacy and non-repudiation in vehicular ad hoc networks, *Proceedings of the IEEE Military Communication Conference-MILCOM'2007*, pp.1-7.
- Theng, Y. & Duh, H. (2008). *Ubiquitous Computing: Design, Implementation and Usability*, IGI Global
- Wang, N.; Huang, Y. & Chen, W. (2008). A novel secure communication scheme in vehicular ad hoc networks. *Computer Communications*, Vol.31, No. 12, pp. 2827-2837
- Want, R. & Pering, T. (2005). System challenges for ubiquitous & pervasive computing, *Proceedings of Int. Conference on Software Engineering - ICSE'05*, pp.9-14
- Washington, L. C. (2008). *Elliptic Curves Number Theory and Cryptography, 2nd ed.*, Chapman & Hall/CRC Press
- Yan, G.; Olariu, S. & Weigle, M. (2008). Providing VANET security through active position detection. *Computer Communications*, Vol. 31, No. 12, pp. 2883-2897
- Yang, L. & Wang, F. Y. (2007). Driving into intelligent spaces with pervasive communications. *IEEE Intelligent Systems*, Vol. 22, No. 1, pp. 12-15.
- Yeun, C. Y.; Lua, E. K. & Crowcroft, J. (2005). Security for emerging ubiquitous networks, *Proceedings of IEEE Vehicular Technology Conference*, Vol.2, pp. 1242-1248.
- Yu, J.Y. & Chong, P.H.J. (2005). A survey of clustering schemes for mobile ad hoc networks. *IEEE Communications Surveys and Tutorials*, Vol. 7, No.1, pp.32-48
- Zhao, J. & Cao, G. (2008). VADD: Vehicle-assisted data delivery in vehicular ad hoc networks. *IEEE Transactions Vehicular Technology*, Vol.57, No.3, pp. 1910-1922.

New Classification of Existing Stream Ciphers

Khaled Suwais and Azman Samsudin
Universiti Sains Malaysia (USM)
Malaysia

1. Introduction

The demand on information security has extensively increased due to the sensitivity of the exchanged information over public communication channels. One of the primary goals of cryptographic systems (cryptosystems) is to help communicators exchanging their information securely. This goal is achieved by cryptographic applications and protocols. Transforming a message (plaintext) to an incomprehensible form (ciphertext) is accomplished by a process known as encryption. In contrast, transforming an encrypted message to its original form is accomplished by a process known as decryption.

In this chapter we focus on one type of symmetric key cryptosystems known as stream ciphers. Stream ciphers are important in securing static and streaming information. Therefore, this chapter intends to present a new classification of stream ciphers based on the keystream generators mechanism. The new classification is based on an extensive review of existing stream ciphers. This review showed that the constructional designs (underlying techniques) of some stream ciphers are similar, resulting in forming new categories for stream ciphers based on those similarities.

The main objectives of this chapter are summarized as follows: to provide comprehensive survey for existing stream ciphers based on their keystream generators, to analyze the security properties of each new category of stream ciphers and to explicate stream ciphers designs through a consistent classification that can assist the development process of new stream ciphers.

The new classification shows that stream ciphers are generally divided into three main categories: software-oriented, hardware-oriented and hybrid-design. This chapter will study the three categories extensively in order to understand the weak and strong points of each category.

2. Stream Cipher: Concept and Definition

Cryptographic systems are divided into two types of systems: Secret-key (Symmetric) and Public-key (Asymmetric) cryptosystems. In the later systems, the sender uses public information of the receiver to send a message securely to the receiver. The receiver then uses

private information to recover the original message. In Secret-key cryptosystems, both the sender and receiver have previously set up secret information in which they use this information for encryption and decryption. Symmetric cryptosystems are further divided into block ciphers and stream ciphers.

The idea of stream ciphers was inspired from the famous cipher called the One-time Pad (Mollin, 2007; Delfs, 2002). This cipher is based on XOR'ing (\oplus) the message bits and the key bits. The One-time pad is defined by Delfs (2002) as shown in Equation 1:

$$E: \{0,1\} \times \{0,1\} \rightarrow \{0,1\}, (m, k) \rightarrow m \oplus k \quad (1)$$

where m and k denote plaintext and keystream bits respectively. The general formulas of encryption and decryption processes are described by Equation 2 and 3 respectively

$$E_{k_i}(m_i) = m_i \oplus k_i = c_i \quad (2)$$

$$D_{k_i}(c_i) = c_i \oplus k_i = m_i \quad (3)$$

Generally, stream cipher uses n -iterations to generate n -successive keystream based on the stream cipher internal state. The review conducted in this study shows that the processing techniques of the internal states of current stream ciphers are vary, where stream ciphers tend to be, in most cases, either hardware-oriented or software-oriented.

3. Stream Ciphers Categories

In contrast to block ciphers, stream ciphers have no standard model for their construction design, which leads cryptographers to construct various models for stream ciphers. This study classifies stream ciphers into categories whereby each category includes stream ciphers that share specific properties. The new classification divides stream ciphers into three main categories: hardware-based stream ciphers, software-based stream ciphers and hybrid designs of stream ciphers. The classification aims to look at stream ciphers from the implementation perspectives. The in-depth classification of hardware-based stream ciphers includes: FCSR/NLFSR-based, clock control-based and LFSR-based stream ciphers. On the other hand, software-based stream ciphers includes: T-function-based, block cipher-based, S-box-based and simple logical and arithmetic operations. The last category, the hybrid designs, includes those stream ciphers which depend on the combination of both hardware and software techniques in their constructional designs. The comprehensive classification of stream ciphers is illustrated by Fig. 1.

3.1 Hardware-Oriented Stream Cipher

The use of hardware implementations was significant in providing the security needed for various cryptographic applications. The widely used hardware implementation, as appears in the literature, relies on the use of LFSRs registers (Bojanic, et al., 2004; Ekdahl, 2003; Canteaut, et al., 2000). However, in this section we briefly introduce LFSRs and analyze the properties of each category and provide some examples on stream ciphers belonging to each category.

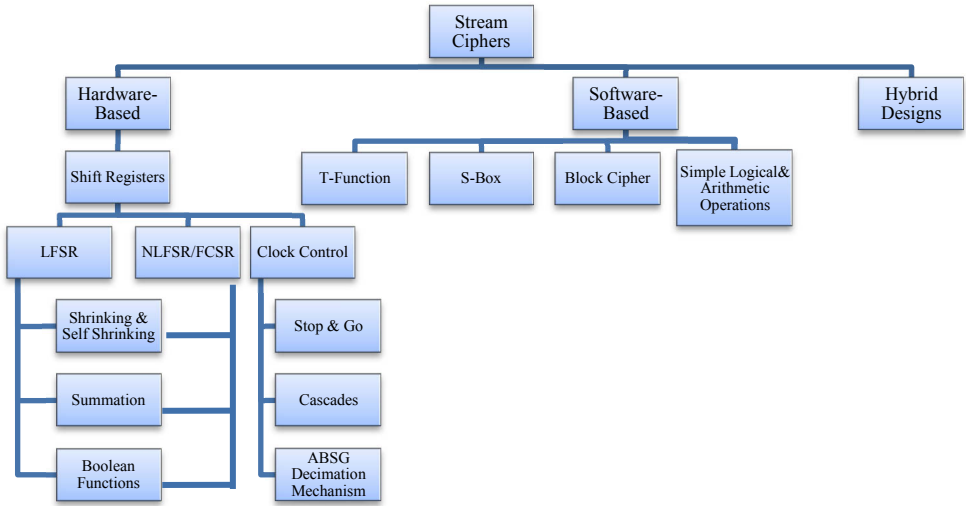


Fig. 1. Stream ciphers classifications

An LFSR is a shift register which is able to hold one symbol at a time and its input is a linear combination of the previous states. The symbols are normally elements from a field F_q , where $q = 2$ refers to the binary fields and 2^w refers to some extension fields of the binary field for a given symbol's size w (Ekdahl, 2003).

Shift register of length ℓ consists of ℓ registers $0, \dots, \ell - 1$ as shown in Fig. 2. Each of these registers is able to hold one symbol, one input and one output. LFSRs rely on system clocks for their operations in which the system clock is responsible for timing all events. With every clocking of the LFSR, the registers read a new symbol from their input, and the symbols move forward from register $\ell - 1$ to register 0 . However, the first register receives the new symbol as a linear combination of the symbols obtained from the previous clocking. Calculating the new symbol is basically determined by the feedback coefficients c_0, c_1, \dots, c_ℓ as referred to in Fig. 2.

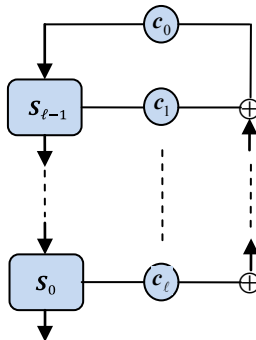


Fig. 2. LFSR of length ℓ

The concept of time clocking is important in LFSR functionality. When the device clocks at time $t > 0$ we obtain a new symbol $S_t \in \mathbb{F}_q$, where S_t is always satisfying the linear recurrence equation found in (Weisstein, 2008; Bolabattin, 2005) as shown in Equation 4:

$$c_0^{-1}S_{t+\ell} - c_1S_{t+\ell-1} - c_2S_{t+\ell-2} - \dots - c_\ell S_t = 0, t > 0 \quad (4)$$

One important feature of using LFSR is its ability to produce an extremely long sequence of linear equation equal to $2n-1$, where n is the number of register elements in the LFSR. Moreover, LFSR is believed to deliver a stream cipher with uniformed distribution of the values generated by the keystream generator. However, the immediate output of LFSR is not acceptable to be used as a keystream since the production of the output value is done in linear fashion. In order to use LFSRs in generating keystream with minimum level of security, non-linear functions have to be added to LFSRs to make the bit production process after each clocking work in non-linear fashion. To achieve this purpose, different techniques have been introduced such as adding some non-linear filters, non-linear updates and irregular clocking to destroy the linearity found in LFSRs.

3.1.1 Shrinking and Self-Shrinking Generator

Coppersmith, et al., (1993) proposed a new generator which consists of two LFSRs and called it as a Shrinking Generator. The shrinking generator is designed as a pseudorandom keystream generator and it is preferred due to the simplicity of its design. The feedback coefficients are represented in polynomial representation. Each one of the LFSR produces a bit stream represented by a_i and b_i produced by LFSR-A and LFSR-B registers respectively to form the keystream K_s . However, shrinking generators are subjected to known-plaintext distinguishing attack which is first introduced by (Ekdahl, et al., 2003). The attack detected some non-randomness in the distribution of the keystream bits. Note that the attacker is required to know the feedback polynomial of A to make this attack feasible.

Self-Shrinking generator is another variant of the shrinking generator concept. The generator rests on a single LFSR instead of using two different LFSRs as in shrinking generators. The procedure of clocking self-shrinking generators works by first clocking two bits from the LFSR, resulting in a pair of bits (a_1, a_2) . If (a_1, a_2) equals to the value $(1, 0)$ or $(1, 1)$, it is taken to produce the pseudorandom bit 0 or 1 respectively. If the pair equals the value $(0, 0)$ or $(0, 1)$, the pair will be discarded because the output will always be a sequence of zeros as reported by Meier, et al., (1994).

Let $\mathbf{a} = (a_0, a_1, a_2, \dots)$ be the output bits of a non-trivially initialized LFSRs of length N . Therefore, \mathbf{a} is a sequence with period $2^N - 1$. With respect to the period of \mathbf{a} , cryptanalysis attack in (Meier, et al., 1994) showed that if the period is at least $2^{N/2}$ and the linear complexity of the construction is $2^{\frac{N}{2}-1}$, attacker can attack the construction in $2^{0.7N}$ steps. Another attack based on a probabilistic approach was introduced by Mihaljevic (1996) and shows that self-shrinking generators can be attacked with complexity 2^{N-1} for any output sequence under certain limitation.

3.1.2 Summation Generator

Rainer Rueppel (1986) introduced a new generator based on the use of LFSRs called the Summation Generator. The idea behind this generator rests on the non-linearity provided by the carry-in integer addition. Rueppel uses this idea to use the output of several LFSRs through an adder with carry which in turn can provide a combination function with good non-linearity and high-order correlation properties (Robshaw, 1995). Rueppel’s summation generator is described as in Equations 5 and 6 (Park, et al., 2005):

$$x_i = a_i \oplus b_i \oplus c_{i-1} \tag{5}$$

$$y_i = a_i b_i \oplus (a_i \oplus b_i) y_{i-1} \tag{6}$$

where a_i is the sequence generated by the first LFSR, b_i is the sequence generated by the second LFSR with the carry initialization value $y_{i-1} = 0$.

In terms of the security of Rueppel’s Summation Generator, the correlation probability of this generator showed that the generator is subjected to correlation attacks (Golic, 1996) since the probability of input-output correlation is $\frac{1}{2}$ (Park, et al., 2005). However, several researchers have tried to improve the security of the summation generator to be used in stream ciphers. One example of stream ciphers using the summation generator is the E0 stream cipher which is used in the Bluetooth protocol (Kitsos, et al., 2003; Galanis, et al., 2005). E0 stream cipher consists of three components: payload key generator (initialize), keystream generator and summation combiner (encoder). However, various cryptanalysis and statistical attacks on E0 were presented in (Lu, et al., 2004), making E0 stream cipher insecure for cryptographic applications. Another example that appears in the literature is a parallelized stream cipher presented in 2002 by Lee and Moon (Lee, et al., 2002). The stream cipher rests on the improvement made on summation generators in (Lee, et al., 2000). Few years later, an algebraic attack against the improved generator was presented in (Han, et al., 2005), making the parallelized stream cipher subject to security vulnerability.

3.1.3 Boolean Function

In mathematics, a Boolean function is defined as a mapping of one or more binary input variables L^k to one binary output variable L . Formally, we write the mapping function as in Equation 7:

$$\beta: L^k \rightarrow L \tag{7}$$

where $L = \{0,1\}$ is the Boolean domain of the Boolean function β , and k is the non-negative integer called the rank of the function. One way of representing Boolean functions with a small number of input variables is by a truth table as illustrated in Table 1.

a_1	a_2	$F(a)$
0	0	0
0	1	0
1	0	1
1	1	0

Table 1. Truth table of the Boolean function $\beta(a_1, a_2) = a_1 a_2 + a_{1i}$

For larger numbers of input variables, it is infeasible to list all the possible values of the truth table. Therefore, we have to use a compact description such as the Algebraic Normal Form (ANF) as shown Equation 8 (Ekdahl, 2003):

$$F = \sum_{u \in F_2^n} \lambda_u \left[\prod_{i=1}^n x_i^{u_i} \right] \quad (8)$$

where $\lambda_u \in F_2$ and $\mathbf{u} = (u_1, u_2, \dots, u_n)$. Another interesting property of Boolean function which attract several cryptographic applications is the balancing of the digits zero and one in the generated sequence. Generally, a Boolean function is said to be balanced if the probability of that function to produce bit 0 or 1, is $\frac{1}{2}$ for all input variables chosen uniformly over F_2^n .

Examples of stream ciphers based on the combination between LFSRs and Boolean functions are found in A5/1 (Biham, et al., 2000) and LILI-128 (Dawson, et al., 2000) stream ciphers. A5/1 was developed in 1987 and later became the most popular stream cipher in most European countries and United States to provide over-the-air communication privacy in GSM cellular telephone standard. The cipher is working in conjunction with three LFSRs (L-A, L-B, L-C) with irregular clocking. The three LFSRs vary in their length, in which the lengths are 19, 22 and 23 for L-A, L-B and L-C respectively. The main idea of A5/1 is to mix the cycled bits generated by the three LFSRs with respect to the irregularity in the clocking process. However, A5/1 seems to be vulnerable to cryptanalysis attacks as presented in (Biryukov, et al., 2000) and (Barkan, et al., 2003).

LILI-128 is another stream cipher which was introduced in 2000 (Dawson, et al., 2000). It uses two binary LFSRs and two Boolean functions to generate a pseudorandom binary keystream. The two functions are evaluated on the current state data and the feedback bits are calculated. Basically, LILI-128 divides the overall work into two subsystems, in which the first subsystem generates some output values and controls the clocking irregularly to control the other subsystem. Nevertheless, several attacks presented in (Jönsson, et al., 2002) and (Tsunoo, et al., 2005) makes LILI-128 insecure.

Finally, there are many other examples on stream ciphers using different techniques (functions, filters, etc) in conjunction with LFSRs to achieve higher security. One example is the stream cipher SNOW (Ekdahl, et al., 2003). SNOW is based on the use of LFSR of the length 16 over an extension to a binary field of 32, feeding a finite state machine. However, SNOW was attacked as presented in (Coppersmith, et al., 2002), and therefore invalidate SNOW to be used for secure applications.

3.1.4 NLFSR and FCSR Registers

Non-Linear Feedback Shift Register (NLFSR) and Feedback with Carry Shift Register (FCSR) are two other types of shift registers used in stream ciphers. The main purpose of these registers is to eliminate and destroy the linearity found in LFSRs. The design of NLFSR applies a non-linear function in the shift register to ensure the non-linearity in the output values from the corresponding shift register. NLFSRs are used in several stream cipher designs such as the Grain stream cipher. Grain was developed in 2004 and submitted to

eSTREAM project for evaluation in 2005 (Hell, et al., 2005). However, Grain was attacked in 2006 by two different cryptanalysts as found in (Maximov, 2006) and (Kucuk, 2006).

FCSRs are similar to LFSR but different in the sense that the elementary addition in FCSR is with propagation of carrier instead of addition modulo 2 as in LFSR. An example of FCSR-based stream cipher is the new stream cipher F-FCSR which was developed recently and submitted for eSTREAM project evaluation (Arnault, et al., 2006). However, F-FCSR was attacked by (Jaulmes, et al., 2006) due to the weaknesses found in the initialization mechanisms as well as lack of entropy of the internal state.

3.1.5 Clock Control

One way of introducing the non-linearity in the generated keystream is by having a shift register clocked irregularly. In other words, the keystream generation is controlled by the varying rate of register clocking. One way of achieving that is by having two or more shift registers such that the clocking of one register is dependent on the other register in some ways. Fig. 3 shows an example of a clock controlled generator called the Altering Step generator where the output of one LFSR controls the other LFSRs.

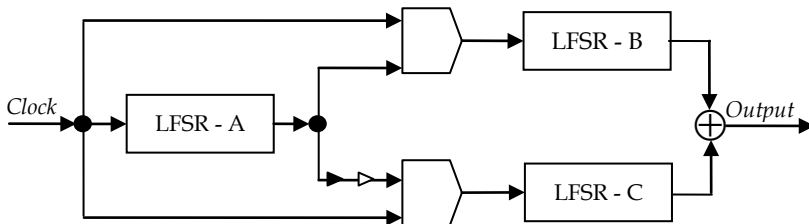


Fig. 3. Alternating step generator

There are various generators that are based on the idea of clock-controlling in shift registers for cryptographic purposes. Some of these generators are: Stop-and-Go, Cascades and ABSG Generators.

Stop-and-Go generator was first introduced in 1985 by Beth and Piper (Beth, et al., 1985). The idea of this generator is to let a control register R-A control the stepping of another register R-B. If the output of R-A is 1, then R-B is clocked. Otherwise R-B is not clocked. The output of R-B is then XORed with the output sequence of a third register R-C. The third register R-C has the same clocking ratio as in R-A. Beth and Piper believe that the stop-and-go generator is secure and immune against cryptanalysis attacks. However, the generator was subjected to efficient cryptanalysis attacks found in (Menezes, et al., 1997) and (Golic, et al., 2003).

Cascade generator is basically an extension of the stop-and-go generator, such that it is still relying on the idea that LFSRs are controlling each other. There are two types of cascades (Robshaw, 1995): The first type allows each register to generate l -sequence and the second type restricts the length of each register to a prime length N with no feedback from any intermediate stage of the register. One example of the cascade stream ciphers is the Pomaranch stream cipher which is based on a Jump Controlled Sequence Generator

(cascade). Unfortunately, Pomaranch was vulnerable to several cryptanalysis attacks found in (Englund, et al., 2007) and (Cid, et al., 2006).

ABSG stream cipher is inspired by the shrinking and self-shrinking generator. Its main purpose was to provide irregularity for the generated keystream bits. Unlike shrinking generators, ABSG operates on a single input variable instead of two. ABSG also differs from the self-shrinking generator in that the production of n -bits of output sequence requires approximate $3n$ -bit of input, while in self-shrinking, the production requires $4n$ -bit of input sequence (Afzal, et al., 2006). The stream cipher DECIM-128 presented in (Berbain, et al., 2005) is based on the use of LFSRs and ABSG decimation mechanism. It is a hardware-oriented stream cipher that handles a secret key of 80-bit length and public initialization vectors of 64-bit. The process of generating keystreams rests on the non-linearity filtered LFSR and the irregular decimation mechanism of ABSG. However, the attack presented in (Wu, et al., 2006) showed that DECIM is suffering from serious flaws in the initialization stage and the keystream generation algorithm stage.

3.2 Software-Oriented Stream Ciphers

In contrast to hardware-based stream ciphers, there are various designs of stream cipher based on bits manipulation (substitution, permutations, etc.), Boolean functions and other alternative designs. These designs of stream ciphers are classified under software-based stream ciphers in which they are not depending on hardware implementations for their security. This section will introduce a variety of stream cipher designs that are associated to different categories. The categorization is based on the mechanisms used in the process of generating keystream sequences used in these ciphers.

3.2.1 T-Function

In 2003, Klimov and Shamir introduced a new type of invertible round function (known as T-Function) by mixing some arithmetic and Boolean operations on full machine words (Klimov, et al., 2003). The name T-function refers to the triangular dependence between the columns of the operands. The function works as a mapping function formulated as in Equation 9:

$$f: \mathbf{B}^{m \times n} \rightarrow \mathbf{B}^{k \times n} \quad (9)$$

where $\mathbf{B} = \{0,1\}$ is represented by a matrix and there is a dependency between the k -th column of the output with the first k set of columns of the input. It was designed to generate pseudorandom values of maximum length. The process of generating $\mathbf{X} = (x_0, x_1, x_2, \dots)$ is described in (Klimov, et al., 2004) and shown in Equation 10:

$$x_i = x_{i-1}^2 \vee \mathbf{C} + x_{i-1} \text{ mod } 2^n \quad (10)$$

where \vee refers to OR operation and \mathbf{C} is used to determine a set of constants defined in the linear equation to hold all the sequences generated by the T-function. Since T-functions are so recent, only few stream ciphers appear in the literatures are based on them. One example is the stream cipher TSC-1 proposed by (Hong, et al., 2005). The proposed cipher is based on a single cycle T-function. TSC-1 works in conjunction with a filter function and 4×4 S-Box.

In general, T-function was subjected to several attacks such as the correlation attack based on the linear approximation of the T-function. The attack was successfully applied on TSC-128 with a complexity of 2^{42} known keystream bits to distinguish it from random (Muller, et al., 2005). The other attack presented in (Künzli, et al., 2005) describes a distinguishing attack on single-word and multi-word T-functions based on the deviation found in the integer differences of consecutive outputs with a complexity of 2^{32} . The importance of T-function comes from the efficiency of implementing it from both hardware and software perspectives. However, it seems that researchers need to put more efforts on developing and enhancing the security aspects of T-function.

3.2.2 S-Box

A substitution box or also known as S-box is an important component of different cryptographic primitives. S-box basically works as a mapping of m input bits into n output bits as visualized in Fig. 4, resulting in an $m \times n$ S-box.

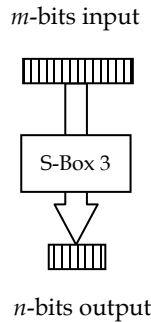


Fig. 4. Input/output mapping using S-Box

The design of S-box comes in two types: fixed and dynamic S-box. Fixed S-boxes rest on pre-computed values calculated in several ways based on the cryptographic component being used. Dynamic S-box are more interesting since the values in the S-box change during the execution. One way of representing S-boxes is by implementing them as table lookups of 2^n entries (Ekdahl, 2003). Another possibility of representing S-box is by calculating the S-box's entries by using a Boolean function as shown in Equation 11:

$$\mathbf{F}(\mathbf{x}) = (\mathbf{F}(x_1), \mathbf{F}(x_2), \dots, \mathbf{F}(x_m)) \quad (11)$$

In this category of stream ciphers, we found few ciphers whose designs are based on S-box. Two examples are discussed here: MUGI and WAKE stream ciphers. MUGI stream cipher was introduced in 2002 as an efficient stream cipher in hardware and software implementations (Watanabe, et al., 2002). MUGI uses a secret key and internal vector of 128-bit length to generate a random string of 64-bit length for each round. The internal state of MUGI consists of two internal states (state a and buffer b) updates by two identical functions (called F-function). The F-function uses three main techniques: key addition, non-

linear S-box and MDS (Maximum Distance Separable) matrix for linear transformation as described in Fig. 5.

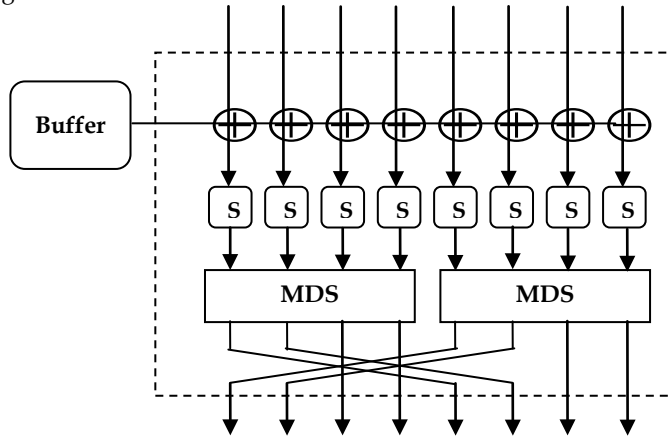


Fig. 5. F-function of MUGI

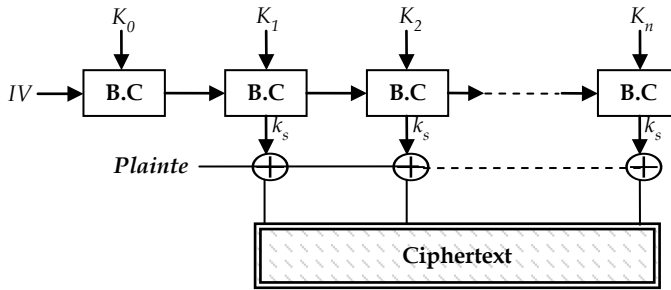
MUGI is not broken yet. However a weakness found in the linear part of MUGI was presented in (Golic, 2004), proved that the real response of the buffer without the feedback from the S-box consists of binary linear recurring sequences with linear complexity and with a very small period of 48 cycles. This theoretical analysis showed that by using the weakness mentioned above, it is possible to use linear cryptanalysis to attack MUGI.

Another example of stream cipher belonging to this category is the WAKE (Word Auto Key Encryption) stream cipher invented by David J. Wheeler (1993). WAKE has a simple structure and performs fast. It produces $4n$ -bit words to be XORed with plaintext to generate ciphertext, or with ciphertext to generate plaintext. The generation of new key depends on the ciphertext produced in the previous round. WAKE uses an S-box of 256 32-bit values with special property where some bytes are obtained from a permutation of all possible bytes, and some other bytes are generated randomly. The S-box of WAKE is not working independently from the overall process of keystream generation; instead it is working as part of function G which uses S-box in conjunction with other shifting operations. However, WAKE was subjected to a chosen plaintext or chosen ciphertext attack, which was fully analyzed in (Pudovkina, 2001). The analysis includes implementing two chosen plaintext attacks on WAKE with a complexity of $10^{19.2}$ and $10^{14.4}$ for the first and second attacks respectively.

It seems that S-box is efficient in providing non-linearity with efficient performance in the internal states of the keystream generators. Designing a cryptographically strong S-box is not easy. Therefore, any misuse of S-box in stream cipher leads to serious security vulnerabilities.

3.2.3 Block Cipher

This is another approach used in the design of stream ciphers. The block cipher is used as a core of the keystream generator of the corresponding stream cipher. The construction of the stream ciphers that belong to this category uses known block ciphers in their keystream generator such as using AES in the stream cipher (Biryukov, 2005). The general structure of stream ciphers based on block cipher is shown in Fig. 6.



B.C.: Block Cipher K: Input Key IV: Initial Value k_s : Keystream

Fig. 6. Stream cipher based on block cipher scheme

Another design philosophy of stream ciphers in this category is based on the Substitution-Permutation Network (SPN) of block ciphers instead of using the components of block ciphers as appeared in Hermes8 stream cipher (Kaiser, 2005). The security of such a design depends on the underlying block cipher (component or technique) that resides at the core of the stream cipher. Up to this day, among the submitted stream ciphers based on block ciphers, LEX and Sosemanuk are the only two ciphers which have moved to the third phase of evaluation of eSTREAM project.

3.2.5 Simple Logical and Mathematical Operations

There are stream ciphers which do not fit into the mentioned categories above. Some of these ciphers are based on bitwise addition and bits rotation operations as in Phelix, SEAL and RC4, while others based on mixing various functions in conjunction with some addition and rotation operations as in Rabbit. In this category we will briefly describe Phelix, SEAL and Rabbit stream ciphers.

- **Phelix Stream Cipher**

Phelix stream cipher (Whiting, et al., 2005) is a high speed stream cipher selected for the software and hardware profiles of eSTREAM project for performance evaluation. Phelix supports an 8-bit to 256-bit length key and 128-bit nonce to generate the keystream bits with embedded MAC code for authentication. The main operations of Phelix are: addition modulo 2^{32} , bitwise XOR and rotation operations. The state of Phelix is broken into two groups: five state words called *active* states which are always participating in updating the internal function and four states called *old* state which is only used in the process of keystream generation.

Since Phelix provides authentication service during transmission, extra processing is done to produce a 128-bit MAC tag to be embedded to the message. Phelix requires 20 rounds in order to produce a single block. The main operations in one block of Phelix is only low-cost operation, in which they are fast in software and hardware implementations. However, Phelix has not moved to the third phase of the eSTREAM project evaluation due to some security vulnerability. Differential-linear attacks presented by Wu and Preneel (2007) showed that with the assumption of reusing the nonce, the key of Phelix can be recovered with complexity 2^{37} chosen plaintext words and $2^{41.5}$ operations. In this attack, the authors showed that Phelix is an insecure stream cipher since recovering the key by reusing the nonce (incorrect use of the nonce) is possible: "In practice an attacker may gain access to a Phelix encryption device for a while, reuse a nonce and recover the key. We thus consider Phelix as insecure" (Wu, et al., 2007).

• Rabbit Stream Cipher

Rabbit is another design of stream ciphers based on iterating a set of coupled non-linear functions – or as the authors called them discretized chaotic maps (Boesgaard, et al., 2003). It uses a 128-bit key and 64-bit initial vector (IV) as input parameters to generate a stream of 128-bit blocks. The encryption is performed by XOR'ing this block with the plaintext block. The inner state of Rabbit consists of 513 bits. The first 512 bits represent 8-state variables (X_0, \dots, X_7) of 32-bit length each and 8-counter variables (C_0, \dots, C_7). The remainder bit is used as a counter carry bit, b . The important part of any stream cipher is the next state function since it is the part that often needs to generate a new keystream. In Rabbit, the next state function is based on function g for mapping two 32-bit inputs to one 32-bit output. Rabbit uses function g to update the inner variables states as shown in Fig. 7.

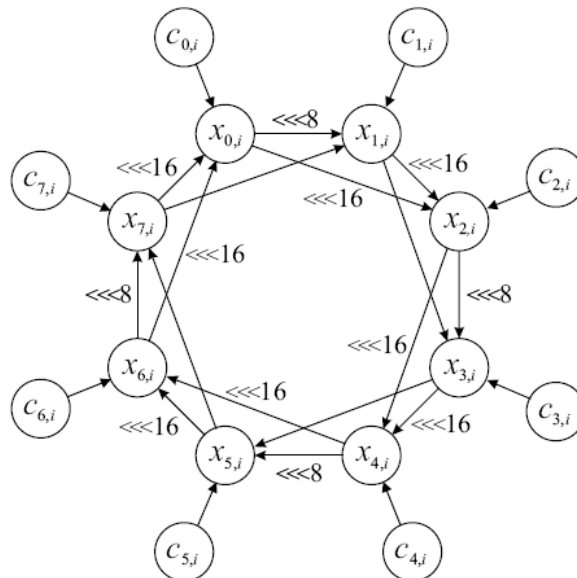


Fig. 7. Updating the inner states of Rabbit (Boesgaard, et al., 2003)

It seems that Rabbit stream cipher is strong against cryptanalysis attacks. It is selected among few other ciphers for further evaluation by eSTREAM project. However, a small bias in the output of Rabbit exists (Aumasson, 2007). Even so, Rabbit is still considered a secure stream cipher since the complexity of the distinguisher is significantly higher than the brute-force attack on the key space, 2^{128} .

- **SEAL Stream Cipher**

SEAL (Software-optimized Encryption Algorithm) is a stream cipher that was designed to work efficiently on software implementation (Rogaway, et al., 1994). SEAL is a length-increasing pseudorandom string mapping function that uses 160-bit encryption key to map (stretch) a 32-bit input length to an n -bit output length. In the pre-processing stage, SEAL uses the hash algorithm SHA-1 (National, 2002) as a part of the table-generation function to stretch the key into a large table. Therefore, part of SEAL's security depends on the security of the used hash algorithm (SHA-1). In terms of the required computation, SEAL requires intensive pre-computation for initializing several large look-up tables with total size approximately 3 KB in size.

In terms of security, SEAL is designed to generate up to 2^{48} bytes of output per input seed. An attack in 1997 showed that this output can be distinguished from random after 2^{34} bytes of output (Coppersmith, et al., 2002). The attack was the reason behind the modification on SEAL, resulting in the modified algorithm SEAL 3.0. In 2001, Fluhrer introduced an attack on SEAL 3.0 that can distinguish the output from random after 2^{44} output bytes (Fluhrer, 2001). It is obvious that SEAL needs to avoid using the same seed after 2^{44} outputs to avoid these types of attacks.

- **RC4 Stream Cipher**

This is yet another important example of stream cipher design. The well known stream cipher is widely used in many security protocols and software applications such as SSL and WEP protocols integrated into Microsoft Windows, Lotus Notes, Apple AOCe, Oracle Secure SQL and many other applications. RC4 (Rivest, 1992) was developed by Ron Rivest in 1987 and the design was kept secret until 1994, until someone anonymously posted it to the Cypherpunks mailing list. The cipher uses a variable key-size with compact code size and it is suitable for byte-oriented processors. The encryption process in RC4 is done by generating a keystream to be XORed with a stream of plaintext to produce a stream of ciphertext.

Generating keystream in RC4 comprises two algorithms: The Key-Scheduling Algorithm (KSA) and the Pseudo-Random Generation Algorithm (PRGA). The KSA algorithm uses a permutation array S of all 256 possible bytes. The two algorithms cooperate with each other as follows: the KSA derives the internal secret state from a variable key size between 40 and 256 bits. PRGA in turn modifies the internal state and produces an output. The initialization process in PRGA sets i and j to 0, and then i is incremented as a counter and j is incremented by adding the value of the permutation array S pointed to by i . The two values of S pointed to by i and j are swapped and the output is resulted by adding $S[i] + S[j]$ modulo 256 as shown in Fig. 8.

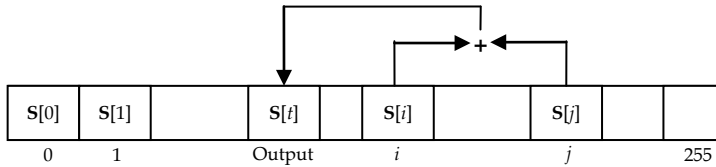


Fig. 8. PRGA round operation

Similar to PRGA, KSA initializes S to the identity permutation and initializes i and j to 0. Sequentially, KSA applies 256 rounds in which i stepped across S and j is updated by adding $S[i]$ to it and the next word of the key. At the present time, RC4 is not recommended for use in new applications. Several weaknesses of the KSA algorithm of RC4 (Fluhrer, et al., 2001) can be summarized in two points. First weakness is the existence of massive classes of weak keys. These classes enable the attackers to determine a large number of bits of KSA output by using a small part of the secret key. Thus, the initial outputs of the weak keys are disproportionately affected by a small portion of key bits. The second weakness rests on a related key vulnerability.

Brute Force attack on RC4 is possible by implementing exhaustive key-searches on Field Programmable Gate Array (FPGAs) using a Network on Chip (NoC) architecture (Couture, et al., 2004). The idea of this attack depends on two components: Key-Checker Unit and the Controller. The latter is responsible for distributing the key space. Key-Checker Unit is used to check each key independently. Therefore, using more than one Checker in a network will provide an adjustable level of parallelism. The research's results shows that RC4 is quite vulnerable to brute-force attack and it is possible to crack RC4 in minutes with a very large FPGA of 500 Checker units in a network.

Other kinds of attacks on RC4 have been presented recently. Results in (Mantin, 2007) showed a statistical bias of the digraphs distribution of the generated stream of RC4. Furthermore, a distinguishing attack was developed based on the statistical bias found in the output sequences (Tsunoo, et al., 2007). This bias is used along with the first two words of a keystream associated with approximately 2^{30} secret keys.

3.3 Hybrid Designs

In this category we discuss other designs of stream ciphers based on a combination of hardware devices and software techniques to achieve the required security. Most of stream ciphers in this category depend on LFSRs as the main component in the core of the stream cipher. The software techniques vary from using T-function as in ABC stream cipher, dynamic permutations as in Polar Bear stream cipher, and look-up tables as in ORYX. In this section we will describe each stream cipher mentioned above and analyze the ciphers' structures and discuss their security strength.

3.3.1 ABC Stream Cipher

ABC is a stream cipher algorithm developed in 2005 (Anashin, et al., 2005) and submitted for eSTREAM project for further evaluation. It deals with a 128-bit key and 128-bit IV. ABC

consists of 38, 32-bit registers. The registers are divided into two groups: 3 registers ($\mathbf{z}^0, \mathbf{z}^1, \mathbf{x}$) are representing the state of ABC, and 35 registers ($\mathbf{d}_0, \mathbf{d}_1, \mathbf{e}, \mathbf{e}_0, \dots, \mathbf{e}_{31}$) represent the constant parameters fed to the cipher. In conjunction with the LFSRs, ABC uses three main functions denoted by A, B and C as shown in Fig. 9.

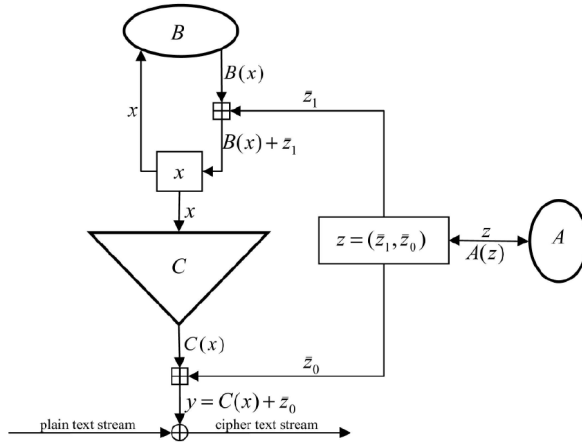


Fig. 9. Functions A, B and C in the keystream generator ABC (Anashin, et al., 2005)

Function A is a linear transformation over the space $GF(2^{64})$, and it is defined by a polynomial characteristic LFSR of length 64. Function B is a T-function with the restrictions that, for the two parameters \mathbf{d}_0 and \mathbf{d}_1 , one must choose these two parameters such that $\mathbf{d}_0 \equiv 1 \pmod{2}$ and $\mathbf{d}_1 \equiv 0 \pmod{4}$ to guarantee that function B is a single cycle map. Lastly, function C is a highly non-linear mapping function (as the authors claimed).

In terms of the security, several attacks on ABC make it fails moving to the third phase of eSTREAM project. Based on the weakness of function C as illustrated in (Khazaei, 2005), a correlation based divide-and-conquer attack was able to find 63-bit of the state by searching 2^{63} possible choices. More specifically, the attack on ABC has a time complexity of 2^{108} to find the whole initial state bits, which is faster than brute-force attack.

A fast correlation attack on ABC was presented in (Zhang, et al., 2006). The attack depends on some weak keys to recover the internal state. Identifying one weak key and recovering the internal state of that key has low computation complexity. The attack is mainly looking for weak keys which were detected in function C for 2^{32} keystream generated from 2^{32} random keys (where each keystream is with 1615 output), the keystream can be distinguished from random. Finding one weak key based on this attack requires $2^{32} \times 1615 \times 4 = 2^{44.7}$ bytes, 2^{45} XOR and 2^{44} addition.

It is obvious that the ABC stream cipher was not strong enough against cryptanalysis attacks. The cipher is considered fast compared to other software-oriented stream ciphers. Nevertheless, choosing cryptographic primitives for secure applications requires more attention on the security side of those primitives. Hence, ABC failed to be considered as a

member of the third phase of eSTREAM project due to the existence of some security vulnerability in its design.

3.3.2 Polar Bear Stream Cipher

Polar Bear was presented in 2005 and submitted to eSTREAM project for evaluation by Johan Hastad and Mats Naslund as reported in (Nada, et al., 2005). The cipher uses one 7-word LFSR (R^0) and one 9-word LFSR (R^1) of length 112-bit and 144-bit respectively. Updating the internal state depends on these two LFSRs along with dynamic permutation of bytes, D_8 . The cipher deals with 128-bit key and up to 32 byte initial vector. Polar Bear uses the Rijndael key schedule for its key schedule algorithm. The initialization process is achieved by taking the expanded key and initial vector, and applies 5 rounds of Rijndael encryption with block length 256. R^0 and R^1 are then loaded by the resulted ciphertext, and D_8 as well as Rijndael S-box are initialized.

The authors of Polar Bear claimed that the cipher is efficient and secure due to the combination of LFSRs with the dynamical permutation. However, a Guess-and-Determine attack presented by Mattsson (Mattsson, 2006) and improved in (Hasanzadeh, et al., 2006), was able to recover the initial states of the registers with time complexity of 2^{79} (by Mattsson attack) and with time complexity of $2^{57.4}$ (by the improved attack). These two attacks showed that the Polar Bear stream cipher is not secure due to the inappropriate usage of the LFSR combined with the dynamical permutations. To counter this attack, it was suggested in (Hasanzadeh, et al., 2006) to initialize the dynamic permutation with an 8×8 key initial vector dependent S-box, provided that the permutation is random to attackers.

3.3.3 ORYX Stream Cipher

ORYX is a stream cipher algorithm that has been proposed for use in North American digital cellular systems. The structure of ORYX is very simple, based on binary LFSRs, S-box (look-up table) and permutation. More specifically, ORYX has four components, three LFSRs of 32-bit length ($LFSR_A$, $LFSR_B$, $LFSR_K$), and an S-box containing a known permutation of the values ranging from 0 to 255, denoted by L . The feedback function for $LFSR_K$ (polynomial characteristic) is defined as in Equation 12:

$$x^{32} + x^{28} + x^{19} + x^{18} + x^{16} + x^{14} + x^{11} + x^{10} + x^9 + x^6 + x^5 + x + 1 \quad (12)$$

while the feedback functions for $LFSR_A$ is defined as in Equation 13:

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1 \quad (13)$$

and finally, $LFSR_B$ is defined as in Equation 14 as follows:

$$x^{32} + x^{31} + x^{21} + x^{20} + x^{16} + x^{15} + x^6 + x^3 + x + 1 \quad (14)$$

The keystream generation is performed by clocking the three LFSRs along with some fixed permutations in order to obtain the high bytes of the current state of each LFSR using a combining function as stated in Equation 15:

$$\mathbf{keystream} = (\mathbf{High8}_k + \mathbf{L}[\mathbf{High8}_A] + \mathbf{L}[\mathbf{High8}_B]) \bmod 256 \quad (15)$$

ORYX is not a secure stream cipher due to the efficient attack presented in (Wagner, et al., 1998). The attack can directly recover the full 96 bits internal state using only 25-27 bytes of known plaintext with time complexity of 2^{16} . Therefore, these results showed that ORYX provides a very low level of security and not suitable for cryptographic applications.

4. Discussion and Conclusion

Increasing the security of the keystream generator is the primary goal for researchers who intend to develop new stream cipher algorithms. The classification presented in this chapter showed that stream ciphers are mainly either software oriented or hardware oriented. In some cases, there are stream ciphers which rely on a combination of hardware devices and software techniques in the design of their keystream generators.

From the security perspective, several stream ciphers (hardware-oriented and software-oriented) are found vulnerable to either cryptanalysis attacks, statistical biased or both. Cryptanalysis attacks on stream ciphers come in two types: hardware-based attacks and software-based attacks. In both types of attacks, attacker tries to extract useful information from the keystream generator in order to obtain the secret key or the plaintext message. Statistical biased such as correlation between keystreams, patterns and randomness are the main issues found in hardware-oriented stream ciphers. On the other hand, the underlying techniques used in software-oriented stream ciphers are found vulnerable to cryptanalysis attacks, due to the relative simplicity of their constructional designs.

Reviewing the constructional designs of stream ciphers leads us to the fact that the keystream generator must be constructed on solid bases. These solid bases can be represented by: linearity-elimination techniques, mathematical hard problems, chaotic behaviours or other secure techniques. The main goal of these new techniques is to provide cryptographic applications with secure stream ciphers against cryptanalysis and statistical attacks.

5. References

- Afzal, M. K., & Masood, A. (2006). Comparative Analysis of the Structures of eSTREAM Submitted Stream Ciphers. In *Proc. The Second International Conference on Emerging Technologies* (pp. 245-250). Peshawar, Pakistan: IEEE-ICET.
- Anashin, V. B., & Kumar, A. (2005, April 29). *ABC: A New Fast Flexible Stream Cipher*. Retrieved May 20, 2008, from The eSTREAM Project: <http://www.ecrypt.eu.org/stream/ciphers/abc/abc.pdf>
- Arnault, F. B., & Lauradoux, C. (2006, January 2). *Update on F-FCSR Stream Cipher*. Retrieved May 26, 2008, from The eSTREAM Project: <http://www.ecrypt.eu.org/stream/papersdir/2006/025.pdf>
- Aumasson, J. (2007, January 2). *On bias of Rabbit*. Retrieved May 30, 2008, from The eSTREAM Project: <http://www.ecrypt.eu.org/stream/papersdir/2007/033.pdf>

- Barkan, E. B., & Keller, N. (2003). Instant Ciphertext-Only Cryptanalysis of GSM Encrypted Communication. In *Advances in Cryptology - CRYPTO 2003* (Vol. 2729 of LNCS, pp. 600-616). Berlin: Springer.
- Berbain, C. B., & Sibert, H. (2005, April 29). *DECIM-128*. Retrieved May 26, 2008, from The eSTREAM Project:
http://www.ecrypt.eu.org/stream/p3ciphers/decim/decim128_p3.pdf
- Beth, T., & Piper, F. (1985). The stop-and-go generator. In *Proc. of the EUROCRYPT 84 workshop on Advances in cryptology: theory and application of cryptographic techniques* (pp. 88 - 92). Paris, France: Springer-Verlag.
- Biham, E., & Dunkelman, O. (2000). Cryptanalysis of the A5/1 GSM Stream Cipher. In *Progress in Cryptology – INDOCRYPT 2000* (Vol. 1977, pp. 43-51). Berlin: Springer.
- Biryukov, A. (2005, April 29). *A new 128 bit key stream cipher : LEX*. Retrieved June 2, 2008, from The eSTREAM Project: <http://www.ecrypt.eu.org/stream/ciphers/lex/lex.pdf>
- Biryukov, A. S., & Wagner, D. (2000). Real Time Cryptanalysis of A5/1 on a PC. In *Proc. Fast Software Encryption*, (pp. 1-18). New York.
- Boesgaard, M. V., & Scavenius, O. (2003). Rabbit: A New High-Performance Stream Cipher. In *Fast Software Encryption* (Vol. 2887 of LNCS, pp. 307-329). Springer.
- Bojanic, S. C., & Taladriz, O. (2004). Stream Cipher Cryptanalysis Based on Edit-Distance: A Hardware Approach. *Tatra Mt. Math. Pub* , 17-29.
- Bolabattin, N. (2005, May 13). *Random Number Generator Using Leap-Forward Techniques*. Retrieved May 21, 2008, from
<http://www.pldesignline.com/showArticle.jhtml?articleID=192200271>
- Canteaut, A., & Filiol, E. (2000). *Ciphertext Only Reconstruction of LFSR-based Stream Ciphers*. LE CHESNAY: Unit'e de recherche INRIA Rocquencourt.
- Cid, C. G., & Johansson, T. (2006). Cryptanalysis of Pomaranch. In *Proc. of IEE Information Security*, 153, pp. 51-53.
- Coppersmith, D. H., & Jutla, C. (2002). Cryptanalysis of stream ciphers with linear masking. In *Advances in Cryptology - CRYPTO'02* (Vol. 2442 of LNCS, pp. 515-532). Springer.
- Coppersmith, D. H., & Jutla, C. (2002). Scream: A Software-Efficient Stream Cipher. In *Fast Software Encryption* (Vol. 2365 of LNCS). Springer.
- Couture, N., & Kent, K. (2004). The Effectiveness of Brute Force on RC4. In *Proc. of the Second Annual Conference on Communication Networks and Services Research* (pp. 333-336). Washington, USA : IEEE Computer Society.
- Dawson, E. C., & Simpson, L. (2000). The LILI-128 Keystream Generator. In *Proc. of First NESSIE Workshop*. Heverlee, Belgium.
- Delfs, H. (2002). *Introduction to Cryptography: Principles and Applications*. Springer.
- Ekdahl, P. M., & Johansson, T. (2003). Predicting the Shriking Generator with Fixed Connections. In E. Biham (Ed.), *Advances in Cryptology - EUROCRYPT2003* (Vol. 2656 of LNCS, pp. 330-344). Springer.
- Ekdahl, P. (2003). *On LFSR Based Stream Ciphers: Analysis and Design*. Lund, Sweden : Lund University.
- Ekdahl, P., & Johansson, T. (2003). A New Version of the Stream Cipher SNOW. In *Selected Areas in Cryptography* (Vol. 2595 of LNCS, pp. 47-61). Berlin: Springer.
- Englund, H. H., & Johansson, T. (2007). Two General Attacks on Pomaranch-Like Keystream Generators. In *Fast Software Encryption* (Vol. 4593 of LNCS, pp. 274-289). Berlin: Springer.

- Fluhrer, S. (2001). Cryptanalysis of the SEAL 3.0 pseudorandom function family. In *Fast Software Encryption* (Vol. 2355 of LNCS, pp. 135-143). Springer.
- Fluhrer, S. M., & Shamir, A. (2001). Weaknesses in the Key Scheduling Algorithm of RC4. In *Selected Areas in Cryptography* (Vol. 2259 of LNCS, pp. 1-24). Berlin: Springer.
- Freier, A. K., & Kocher, P. (1996). *The SSL Protocol Version 3.0*. Retrieved January 15, 2008, from <http://wp.netscape.com/eng/ssl3/ssl-toc.html>.
- Galanis, M. K., & Goutis, C. (2005). Comparison of the Hardware Implementation of Stream Ciphers. *The International Arab Journal of Information Technology*, 2 (4), 267-274.
- Golic, D., & Menicocci, R. (2003). Edit probability correlation attacks on stop/go clocked keystream generators. *Journal of cryptology*, 16 (1), 41-68.
- Golic, J. (2004). A Weakness of the Linear Part of Stream Cipher MUGI. In *Fast Software Encryption* (Vol. 3017 of LNCS, pp. 178-192). Berlin: Springer.
- Golic, J. (1996). Correlation properties of a general combiner with memory. *Journal of Cryptology*, 111-126.
- Han, D., & Lee, M. (2005). An algebraic attack on the improved summation generator with 2-bit memory. *Information Processing Letters*, 93 (1), 43 - 46.
- Hasanzadeh, M. S., & Khazaei, S. (2006). *Improved Cryptanalysis of Polar Bear*. Retrieved May 29, 2008, from The eSTREAM Project: <http://www.ecrypt.eu.org/stream/papersdir/084.pdf>
- Hell, H. J., & Meier, W. (2005, April 29). *Grain - A Stream Cipher for Constrained Environments*. Retrieved May 26, 2008, from The eSTREAM Project: <http://www.ecrypt.eu.org/stream/ciphers/grain/grain.pdf>
- Helleseeth, T. J., & Kholosha, A. (2006, January 2). *Pomaranch - Design and Analysis of a Family of Stream Ciphers*. Retrieved May 27, 2008, from The eSTREAM Project: <http://www.ecrypt.eu.org/stream/papersdir/2006/008.pdf>
- Hong, J. L., & Han, D. (2005). A New Class of Single Cycle T-Functions. In *Fast Software Encryption* (Vol. 3557 of LNCS, pp. 68-82). Berlin: Springer.
- Jaulmes, É., & Muller, F. (2006). Cryptanalysis of the F-FCSR Stream Cipher Family. In *Selected Areas in Cryptography* (Vol. 3897 of LNCS, pp. 20-35). Berlin: Springer.
- Jönsson, F., & Johansson, T. (2002). A fast correlation attack on LILI-128. *Information Processing Letters*, 81 (3), 127 - 132.
- Kaiser, U. (2005, April 29). *Hermes Stream Cipher*. Retrieved May 20, 2008, from eSTREAM PHASE 2: <http://www.ecrypt.eu.org/stream/ciphers/hermes8/hermes8.pdf>
- Khazaei, S. (2005). *Divide and conquer attack on ABC stream cipher*. Retrieved May 15, 2008, from eSTREAM, ECRYPT Stream Cipher Project: <http://www.ecrypt.eu.org/stream>
- Kitsos, P. N., & Koufopavlou, O. (2003). Hardware Implementation of Bluetooth Security. *IEEE Pervasive Computing*, 2 (1), 21-29.
- Klimov, A., & Shamir, A. (2003). A New Class of Invertible Mappings. In *Cryptographic Hardware and Embedded Systems - CHES 2002* (Vol. 2523 of LNCS, pp. 470-483). London, UK: Springer.
- Klimov, A., & Shamir, A. (2004). New Cryptographic Primitives Based on Multiword T-Functions. In *Fast Software Encryption* (Vol. 3017 of LNCS, pp. 1-15). Berlin: Springer.
- Kucuk, O. (2006, July 16). *Slide Resynchronization Attack on the Initialization of Grain 1.0*. Retrieved May 25, 2008, from The eSTREAM Project: <http://www.ecrypt.eu.org/stream/papersdir/2006/044.ps>

- Künzli, S. J., & Meier, W. (2005). Distinguishing Attacks on T-Functions. In *Progress in Cryptology – Mycrypt 2005* (Vol. 3715 of LNCS, pp. 2-15). Berlin: Springer.
- Lee, H., & Moon, S. (2000). On an improved summation generator with 2-bit memory. *Signal Processing*, 80 (1), 211-217.
- Lee, H., & Moon, S. (2002). Parallel stream cipher for secure high-speed communications. *Signal Processing* (82), 259-265.
- Lu, Y., & Vaudenay, S. (2004). Cryptanalysis of Bluetooth Keystream Generator Two-Level E0. In *Advances in Cryptology - ASIACRYPT 2004* (Vol. 3329 of LNCS, pp. 483-499). Berlin: Springer.
- Mantin, I. (2007). Predicting and Distinguishing Attacks on RC4 Keystream Generator. In *Advances in Cryptology* (Vol. 3494 of LNCS, pp. 491-506). Berlin: Springer.
- Mattsson, J. (2006). *A Guess-and-Determine Attack on the Stream Cipher Polar Bear*. Retrieved May 10, 2008, from The eSTREAM Project:
<http://www.ecrypt.eu.org/stream/papersdir/2006/017.pdf>
- Maximov, A. (2006). Cryptanalysis of the "Grain" family of stream ciphers. In *Proc. of the 2006 ACM Symposium on Information, computer and communications security* (pp. 283 - 288). Taipei, Taiwan: ACM.
- Meier, W., & Staffelbach, O. (1994). The Self-Shrinking Generator. In *Eurocrypt 94* (Vol. 950 of LNCS, pp. 205-214). Springer.
- Menezes, A. O., & Vanstone, S. (1997). *Handbook of Applied Cryptography*. Boca Raton, FL: CRC Press.
- Mister, S., & Adams, C. (1996). Practical S-Box Design. *Workshop on Selected Areas in Cryptography (SAC '96)* (pp. 61-76). Philadelphia, Pennsylvania: ACM.
- Molland, H., & Hellesteth, T. (2005). Alinear weakness in the Klimov-Shamir T-function. In *Proc. International Symposium on Information Theory, ISIT 2005*. (pp. 1106 - 1110). Adelaide, Australia: IEEE.
- Mollin, R. A. (2007). *An Introduction to Cryptography* (2nd Edition ed.). (K. H. Rosen, Ed.) Boca Raton: Chapman & Hall/CRC.
- Muller, F., & Peyrin, T. (2005). Linear Cryptanalysis of the TSC Family of Stream Ciphers. In *Advances in Cryptology - ASIACRYPT 2005* (Vol. 3788 of LNCS, pp. 373-394). Berlin: Springer.
- Nada, J., & Naslund, M. (2005). *The Stream Cipher Polar Bear*. Retrieved May 10, 2008, from The eSTREAM Project:
<http://www.ecrypt.eu.org/stream/ciphers/polarbear/polarbear.pdf>
- National, S. A. (2002). *Announcing the Secure Hash Standard*. Federal Information Processing Standards Publications 180-2.
- Park, M., & Park, D. (2005). A New Stream Cipher Using Two Nonlinear Functions. In *Computational Science and Its Applications - ICCSA 2005* (Vol. 3481 of LNCS, pp. 235-244). Berlin: Springer.
- Pudovkina, M. (2001). *Analysis of chosen plaintext attacks on the WAKE Stream Cipher*. Retrieved May 29, 2008, from eprint: <http://eprint.iacr.org/2001/065.pdf>
- Rivest, R. (1992). *The RC4 Encryption Algorithm*. RSA Data Security Inc.: Document No. 003-013005-100-000000.
- Robshaw, M. (1995). *Stream Ciphers*. CA: RSA Laboratories.
- Rogaway, P., & Coppersmith, D. (1994). A software-optimized encryption algorithm. In *Fast Software Encryption* (Vol. 809 of LNCS, pp. 56-63). Springer.

- Stalling, W. (2003). *Cryptography and network security: principles and practice* (3rd ed.). New Jersey: Prentice Hall.
- Tsunoo, Y. K., & Suzuki, T. (2007). A Distinguishing Attack on a Fast Software-Implemented RC4-Like Stream Cipher. *IEEE Trans. on Information Theory*. 53, pp. 3250-3255. IEEE Computer Society.
- Tsunoo, Y. S., & Minematsu, K. (2005). Shorter Bit Sequence Is Enough to Break Stream Cipher LILI-128. *IEEE Trans. on Information Theory*. 51 (12), pp. 4312-4319. IEEE Computer Society.
- Wagner, D. S., & Schneier, B. (1998). Cryptanalysis of ORYX. In *Selected Areas in Cryptography* (Vol. 1556 of LNCS, pp. 296-305). Springer.
- Watanabe, D. F., & Preneel, B. (2002). A New Keystream Generator MUGI. In *Fast Software Encryption* (Vol. 2365 of LNCS, pp. 179-194). Berlin: Springer.
- Weisstein, E. W. (2008). *Gram-Schmidt Orthonormalization*. Retrieved July 20, 2008, from MathWorld--A Wolfram Web Resource: <http://mathworld.wolfram.com/Gram-SchmidtOrthonormalization.html>
- Wheeler, D. (1993). A Bulk Data Encryption Algorithm. In *Fast Software Encryption, Cambridge Security Workshop* (Vol. 809 of LNCS, pp. 127 - 134). London, UK: Springer.
- Whiting, D. S., & Muller, F. (2005, April 29). *Phelix: Fast Encryption and Authentication in a Single Cryptographic Primitive*. Retrieved May 12, 2008, from The eSTREAM Project: <http://www.ecrypt.eu.org/stream/ciphers/phelix/phelix.pdf>
- Wu, H., & Preneel, B. (2007). Differential-Linear Attacks Against the Stream Cipher Phelix. In *Fast Software Encryption* (Vol. 4593 of LNCS, pp. 87-100). Berlin: Springer.
- Wu, H., & Preneel, P. (2006). Cryptanalysis of the Stream Cipher DECIM. In *Fast Software Encryption* (Vol. 4047 of LNCS, pp. 30-40). Berlin: Springer.
- Zenner, E. (2004). *Cryptanalysis of LFSR-based Pseudorandom Generators - a Survey*. Reihe Informatik.
- Zhang, H. L., & Wang, X. (2006). *Fast Correlation Attack on Stream Cipher ABC v3*. Retrieved May 18, 2008, from <http://www.ecrypt.eu.org/stream/papersdir/2006/049.pdf>

Smart Web Based Programming Contests Management Tool

Ahmed Bentiba, Mohamed J. Zemerly and Mohamed Al Mansoori
*Khalifa University of Science, Technology and Research, Sharjah
United Arab Emirates*

1. Introduction

Serious programming languages competitions world wide use the famous *Programming Contest Control System*, PC², developed at the California State University, Sacramento (CSUS, 2009)

We were involved during several years in local and regional ACM International Collegiate Programming (ICPC). We have participated at all levels: contestant, coaches and chief judge (KUSTAR, 2006 & 2007).

We found that PC² system has many limitations and weaknesses. PC² interface is limited to English language. It is mainly used for programming language contests and it is used in local area competitions in one center for example. Although it can be used in different centers, it cannot be used behind firewalls: "in a multi-site contest, every machine running a PC² server must be able to communicate via TCP/IP with the machines running PC² servers at every other site. In particular, there must not be any firewalls which prohibit these communication paths; the system will not operate if this communication is blocked" (CSUS, 2009).

PC² system is restricted only to programming contest types and it is limited to Pascal C/C++ and Java. Recently, Pascal has been dropped as a World Finals Language (CSUS, 2009).

For these reasons and many others, we decided to design an alternative to this system. We called it Wide Area Contests System, WACS (see Figures 1 and 2) to allow people from different cities or countries to participate in different contests not only programming language contests.



Fig. 1. WACS Login Page

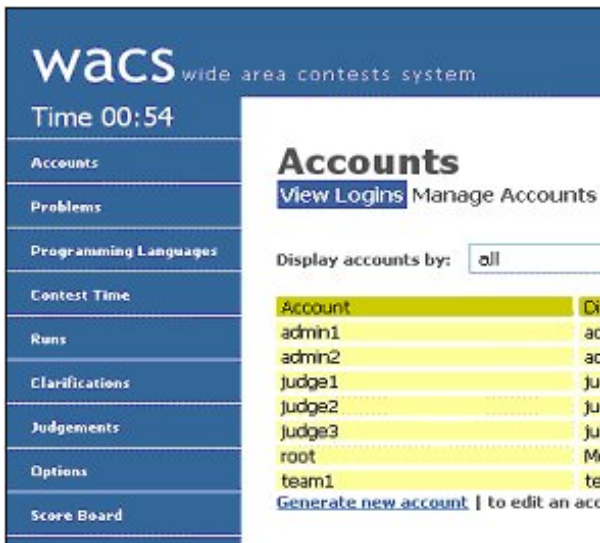


Fig. 2. WACS Web Interface

WACS is developed using different web languages and technologies such as PHP (Lerdorf, 1995), CSS (Lie, 1994; McFarland, 2006), JavaScript (Eich, 2005; Goodman & Morrison, 2004) and Ajax (Garrett, 2005; Zakas et al., 2007). It uses MySQL relational database system (Widenius & Axmark, 1994; Schwartz, 2008) to store its information rather than plain files. It is then more flexible and easier to add modules to WACS than to PC². Connecting to a contest on WACS is just as easy as to browse web pages. All what teams need is the contest’s web address, login names and passwords. Figure 3 shows WACS web admin page.

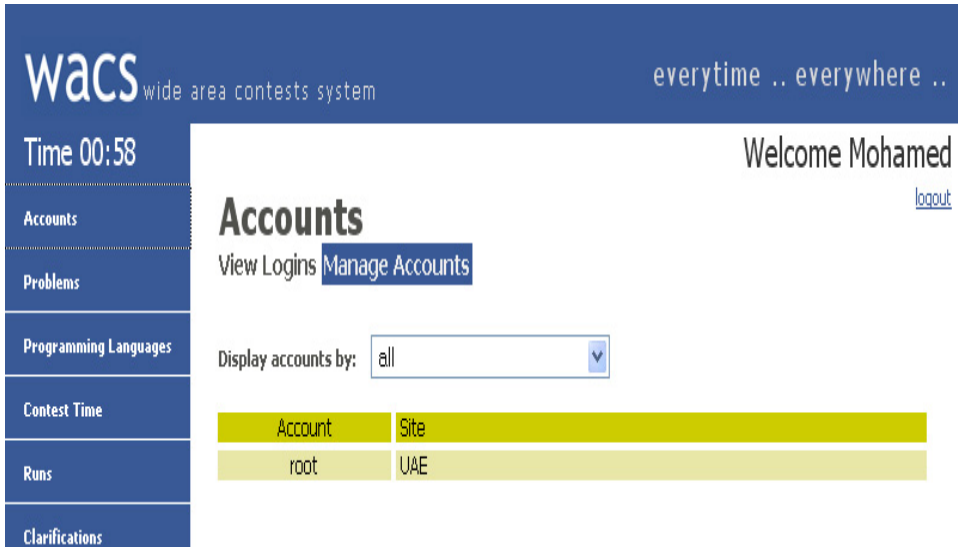


Fig. 3. WACS Admin Page

2. System Description

Although the original idea in developing the WACS system was to allow teams from different places to participate in different contests, WACS is implemented with a focus on programming languages contests as in Figure 4. However, other contest types, such as math, physics, Arabic, history and so forth can be easily plugged to the system as long as they are written using web languages and technologies.



Fig. 4. Computer Languages

The system has three categories of users, which are administrator, judges and contestants or teams. The administrator has full control for the system. He/she is responsible for:

1. adding, updating and deleting accounts,
2. adding, updating and deleting problems or questions to solve,
3. adding or modifying programming languages to the system,
4. set the contest time, etc.

The system allows the contestants (teams) to interact with the judges during the contest through messages to request clarification and submit their answers to the problems. The judges can communicate with each team through messages to answer their requests. The judges can read the programs codes sent by teams and they can run them to evaluate them. The WACS system is able to run each team program code. Finally, the system provides a scoreboard that shows the results during the contest period with automatic update using the Ajax programming language without any manual page reload, as it is the case in PC². Ajax is originally considered an acronym for Asynchronous JavaScript + XML, the term is now used simply to encompass all the technologies that allow a browser to communicate (CSUS, 2009).

3. System Architecture

The WACS system is divided into two major parts: the base of the system and the contest type. The base of the system is where the WACS system allows people in:

- the same room or different rooms,
- in the same building or in different buildings,
- in the same city or in different cities,
- and in the same country or in different countries

to participate in a contest. It consists of an Apache web server (Apache, 2009) and a MYSQL database server (Schwartz et al., 2008) with PHP engine (Converse & Park, 2002).

On the client side, each educational institution can participate with one or more teams. Each team is composed of three students supervised by one or more coaches.

The contest itself is designed and supervised by the contest steering committee. During the contest, the responsibility is shifted from the administrators to the judges. The administrator will provide the judges with the technical support they need to run the competition smoothly. Figure 5 shows the main parts of the system's components.

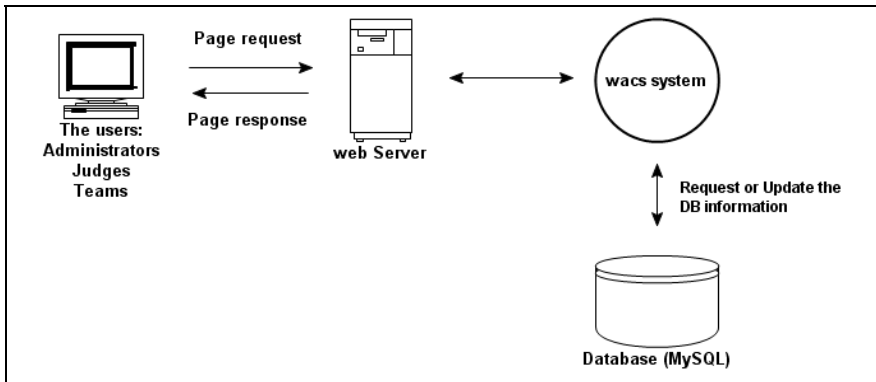


Fig. 5. WACS Components

Users communicate and interact with each other through the web server. Every category of users has a different web page to access the WACS. Any action performed in WACS is recorded within the MySQL database system, such as logins, submitting problems' solutions, etc.. During the contest, each team can interact with the judges through the WACS system only by submitting their solutions and they may ask for clarifications or if they have any doubt regarding any contest question. The judges also can send contestants feedbacks or special messages through WACS only. There is no physical contact between teams and judges during the contest and between contestants and their coaches. The block diagram in Figure 6 shows the interactions between the three categories of users in the system.

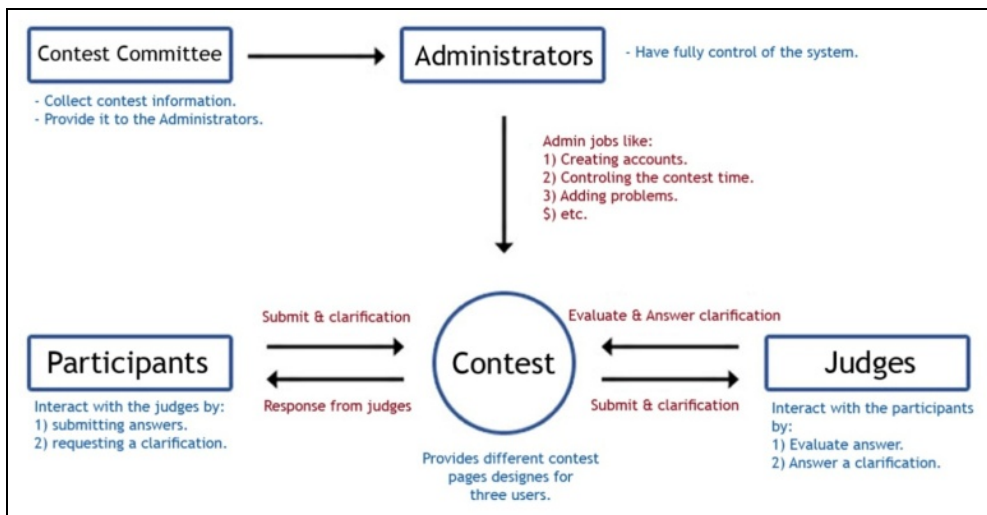


Fig. 6. Interactions between WACS users.

Figure 7 shows the entity relationship between various WACS tables. The details of these tables are not covered in this chapter.

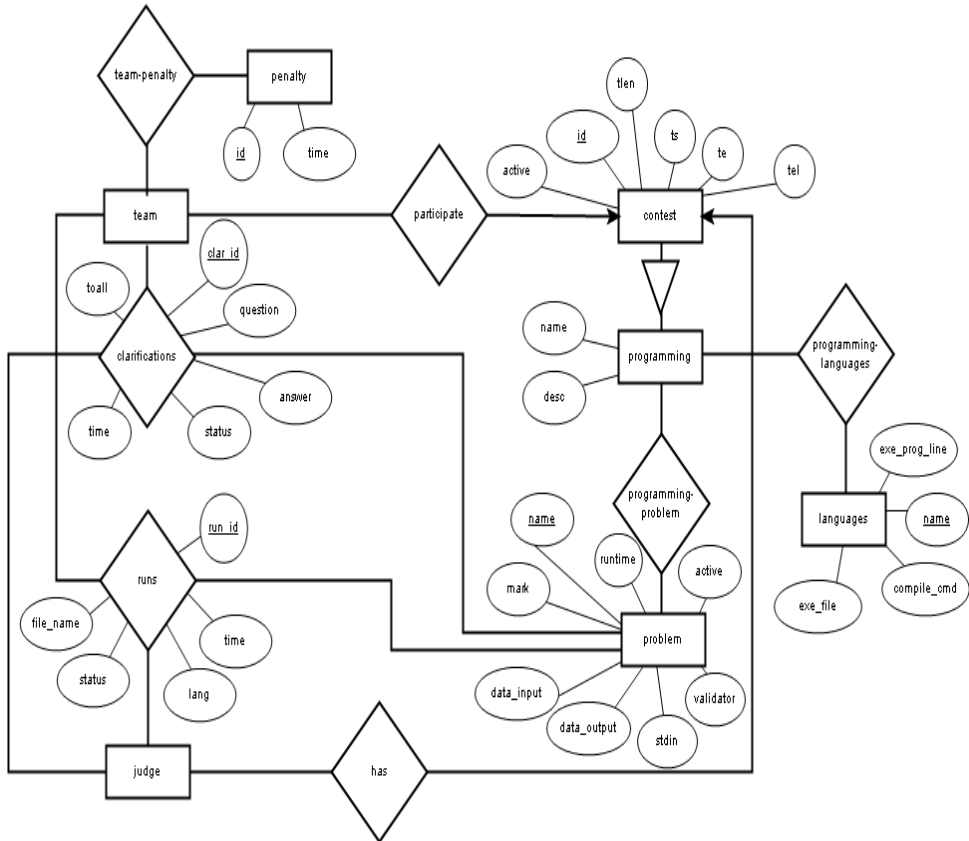


Fig. 7. Entity Relationship Diagram

Figure 8 shows the state-transition of the WACS system. The figure is self-explanatory.

3. Programming contest

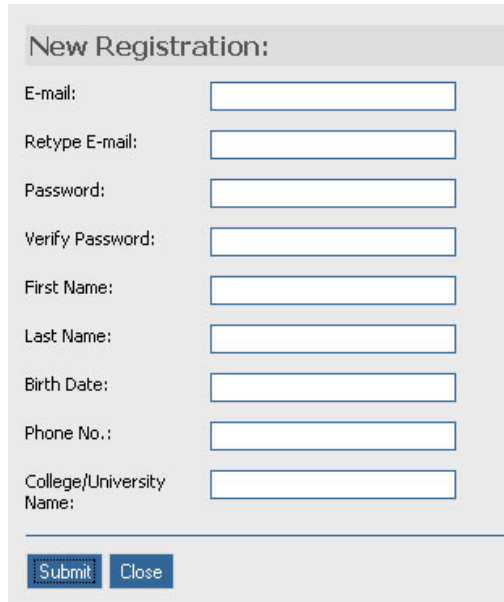
The WACS system is implemented to allow only coaches to register teams. Unregistered coaches, who have teams to participate in the contest, can register and then create accounts for their teams. Coaches have to fill standard forms as shown in Figure 9 to collect information such as names, passwords and email addresses and so on.



Fig. 8. State-transition diagram

The WACS stores the information in the database after checking whether it is there or not and sends an email to the coaches to activate their accounts.

The teams can submit their source code, once they finished solving any problem, via Submit window as shown in Figure 10. It provides the team with two options test the program before uploading it and submit it to the judges.



New Registration:

E-mail:

Retype E-mail:

Password:

Verify Password:

First Name:

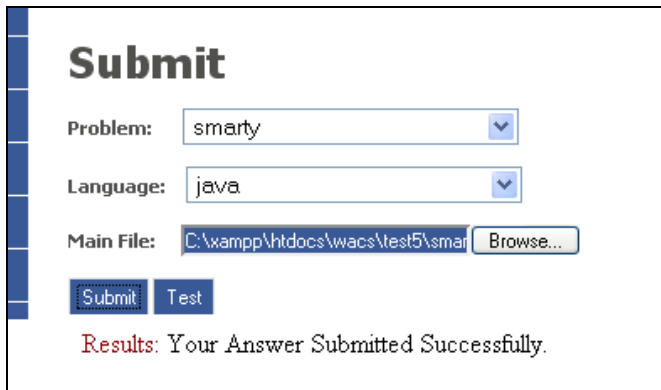
Last Name:

Birth Date:

Phone No.:

College/University Name:

Fig. 9. Registration Form



Submit

Problem:

Language:

Main File:

Results: Your Answer Submitted Successfully.

Fig. 10. Submit and Test Page

The WACS administrator can add or remove problems. For each problem, he/she has to set the mark for it and to insert the input and output files into the database. WACS has a script that allows judges to compare the teams' results (outputs) with the results stored in the database for each problem. Depending on the results of these comparisons, the judges send appropriate messages to any team that submitted any source code for evaluation. The administrator manages the time of the contest. He/She can start, stop or reset the contest time. Finally, WACS has a scoreboard as shown in Figure 11 that allow users to view the teams' ranks, the number of tries and the number of problems solved by each team and the time taken to solve these problems. It fetches the information from the database and it is refreshed automatically using Ajax techniques (Asleson & Schutta, 2006).

wacs wide area contests system		everytime .. everywhere ..													
Time 00:47		Welcome Mohamed logout													
Accounts		Score Board													
Problems		Rank	Name	Solved	Time	I	H	G	F	E	D	C	A	B	Total alt/solv
Programming Languages		1	team5	0	0	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/0
Contest Time		2	team4	0	0	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/0
Runs		3	team3	0	0	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/0
Clarifications		4	team2	0	0	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/--	0/0
Judgements		5	team1	0	0	0/--	0/--	0/--	0/--	0/--	0/--	0/--	1/--	0/--	0/0
Options		Submitted/1st Yes/Total Yes				0/--/0	0/--/0	0/--/0	0/--/0	0/--/0	0/--/0	0/--/0	1/--/0	0/--/0	1/0

Fig. 11. Contestants Score Board

3. Conclusion

This chapter describes the WACS system as an alternative to the famous *Programming Contest Control System, PC²*. The implementation of the WACS system is mainly focused on programming languages contests. However, to add any other contest type is an easy task because of the flexibility of web programming languages and technologies. We used extensively, during the development of the WACS, open-source applications, web languages and technologies such as XHTML, CSS, PHP, Apache, MySQL, and Ajax. Since WACS should be installed on a server (running Linux or Windows) and without any component on client machines, it will reduce the burden for educational institutions or any participant without IT experts. All the configurations files, applications and servers (web, database) reside on the main server, which should be in the institution who hosts the competition. The only requirement needed in order to connect to WACS is a web-browser.

The WACS system can be used inside computer labs, where students are given one or more problems to solve in a given period of time let say one or two hours. This activity will encourage students to work in a competitive environment. For the lab supervisors the code marking will be very easy and quick.

Also students can use the WACS system in their free time such as breaks or weekends to improve problem solving skills and to practice in real competition environment.

4. Future work

The WACS system needs to be tested in large group of teams and between different cities or countries to study the system limitations and performance. Regarding the performance of the WACS system, one should consider the compilation and the execution of programs to be put on a different physical server than the WACS web server. We should have more than one server in the system: One web server to host the WACS system, and at least one more server dedicated to compile the code files, execute the programs on other machines, and get the results from them. Some features can be added to the system, such as:

1. Security for login: right now, the passwords are sent to the system in clear text. It should be encrypted

2. The compilation of users' programs and running them are done with root/administrator privileges, which is not secure at all, since users, can send malicious code to damage the whole system. The solution to this is to use the system as unprivileged users such as nobody in Linux to limit the system damage.
3. Mirroring the system
4. The surveillance system: Cameras should be added in each center and should be controlled and monitored by the judges. One camera to monitor each team and several cameras to cover the whole center. If the camera system fails the center is withdrawn from the competition. As an alternative to cameras, each center or educational institution has to send one judge or staff to another center to supervise the competition. The educational institutions will cover the travel and hotel expenses of only this staff instead of all participants and their coaches.
5. The WACS system is implemented and tested for programming languages and can be extended easily for other subjects such as math, physics, Arabic or any other type to written tests.
6. The WACS interface is in English language, but we hope to implement also translations to other languages such Arabic, French and others.
7. As it is implemented now, only the administrator can start contests, add problems and users. It would be nice to allow some privilege users such lecturers or coaches to start their contests, to add their problem sets and to create their users.

5. References

- Apache Group (2009). Available [Online] <http://httpd.apache.org/>
- Asleson, R., Schutta, N. T. (2006). Foundations of Ajax, Apress.
- Brendan, E. (2005). Available [Online] <http://en.wikipedia.org/wiki/JavaScript>
- Converse, T. CSUS (2009). PC2 Home Page. Available [Online] <http://www.ecs.csus.edu/pc2/>
- Garrett, Jesse James (2005). Available [Online] [http://en.wikipedia.org/wiki/Ajax_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming))
- Goodman D., Morrison M. (2004). JavaScript Bible, Fifth Edition Wiley.
- KUSTAR (2006). National Programming Contest 2006. Available [Online] <http://www.euc.ac.ae/eceportal/ece/npc2006.htm>
- KUSTAR (2007). National Programming Contest 2007. Available [Online] <http://www.euc.ac.ae/eceportal/ece/npc07.htm>
- Lerdorf, Rasmus (1995). Available [Online] <http://en.wikipedia.org/wiki/Php>, <http://www.php.net>
- Lie, Håkon Wium 1994, Available [Online] <http://en.wikipedia.org/wiki/CSS>
- McFarland, D. (2006). CSS: The Missing Manual, Pogue Press 1st edition.
- Park, J. (2002). PHP Bible, Wiley, 2nd edition.
- Schwartz, B., Zaitsev, P., Tkachenko, V., Zawodny, J. (2008). High Performance MySQL: Optimization, Backups, Replication, and More, O'Reilly Media 2nd edition.
- Sklar, D., Trachtenberg, A. (2002). PHP Cookbook O'Reilly Media.
- Zakas, N. C. , McPeak, J., Fawcett, J. (2007). Professional Ajax, Wrox, 2 edition.
- Widenius, M., Axmark, D. (1994). Available Online : <http://en.wikipedia.org/wiki/Mysql>

Intelligent Exploitation of Cooperative Client-Proxy Caches in a Web Caching Hybrid Architecture[†]

Maha Saleh El Oneis, Mohamed Jamal Zemerly and Hassan Barada
Khalifa University of Science, Technology, and Research
United Arab Emirates

1. Introduction

The technological evolution witnessed by the world today has made the migration of services and information to the World Wide Web (WWW) faster and easier. This, in turn, made the number of Internet users increase exponentially. Such increase in number of users, and the demand of information and services resulted in a variety of problems that affected the user's comfort in using and retrieving information from the Web. Despite all the advanced technologies used today, two main problems are still faced which are *server overloading* and *network congestion*. Network congestion can occur when a network link is carrying too much data that would affect its quality of service. Server overloading happens when the server receives more service requests than it can handle. Many researchers have tackled these issues since the early 90's and some helpful solutions have been implemented. One of the most effective solutions that alleviate server overload and reduce network congestion is *web caching*. Web caching is the process of saving copies of content (obtained from the Web) closer to the end user, in order to reduce bandwidth usage, prevent server overload, as well as reducing the user's perceived latency. These studies have resulted in the development of a web caching notion which can be implemented at three levels. Level1 (L1) cache is known as the client caching which takes place at the browser level. Level2 (L2) takes place at the proxy level while Level3 (L3) is the cooperation of the proxies in sharing cached objects among the cooperation set (Dykes & Robbins, 2002). Researchers have agreed that caches on the client browser and proxy level can significantly improve performance (Abrahams et al., 1995). In addition, many studies encouraged the broad use of web caching within organizations that provide internet service to users (Korupolu & Dahlin, 1999; Gadde & Robinovich, 1999; Wolman & Voelker, 1999; Lee et al., 2001). Such studies helped in considering the possibility of constructing a large number of web caches from cooperative proxies and client caches (Zhu & Hu, 2003) to introduce a new cooperation level.

[†] This project is funded by Intel Innovation Center, Dubai, U.A.E.

A range of studies agreed on the benefits of web caching and its major contribution to Internet services. Still the rapid growth of internet traffic and users has made us witness rapid improvements on the broadband services. Nowadays, Internet Service Providers (ISPs) are offering better broadband networking technologies, but still many residential and small-business users are using low-bandwidth connections. Any near promise of the availability of such broadband technologies for users in rural areas is still uncertain because of the associated high cost. However, even with the availability of high bandwidth, there are types of information such as multimedia that always demand more bandwidth. For example when YouTube became very popular, one of the Internet Service Providers (ISP) had huge increase in the amount of information entering the network and an increase in the user's perceived latency. When the problem was observed closely, the ISP discovered that 1Gb/s in the network was consumed by only one website, YouTube.com. In addition to the obvious benefits of web caching, some of the important properties desired in a web caching scheme are fast access, robustness, transparency, scalability, efficiency, adaptivity, stability, load balanced, ability to deal with heterogeneity, and simplicity (Wang, 1999). Our area of interest in building a new hybrid web caching architecture is to reduce the client latency period in retrieving WWW information in rural areas as well as improving the performance of the broadband technology. This architecture explores and benefits from the free space offered by the client's caches when they are connected to the internet, and reduces the load on the upper tier (proxies & web) servers.

With the exponential growth of the internet, a single web cache is expected to become a hot spot. If the solution is to add more hardware, we can expect a point where no hardware is sufficient enough, and the management of such number of extra hardware is a burden in different aspects.

2. Related Work

Ever since web caching has been found as a solution for network congestion and server overloading, different caching architectures were proposed to ease the process of delivering the requested data through inter-cache cooperation. The next sections discuss some of the most common web caching architectures proposed in recent years. We classify these architectures into hierarchical architecture, distributed architecture, and hybrid architecture.

2.1 Hierarchical Caching Architecture

The idea behind constructing a hierarchical cache is to arrange a group of caches in a tree-like structure and allow them to work together in a parent-child relationship to fulfil the requested objects by the client. If a hierarchical structure is arranged properly, the hit ratio can be increased significantly (Wang, 1999).

In a hierarchical caching architecture, caches are placed at various levels of the network. As shown in Figure 1, there is a client's cache, institutional cache, regional cache, national cache, and at the top is the original server. When a client requests a page, it first checks its browser cache. If the request is not fulfilled, then it is forwarded to the institutional cache. If the request is not satisfied by the institutional cache, then it is passed to the regional cache. If the request is not found at the regional cache, then it is redirected to the national cache. The national cache forwards the request to the original server if it cannot fulfill the request.

When the object is found in a cache or the original server, it travels down the hierarchy and leaves a copy of the object in each caching level in its path to the client.

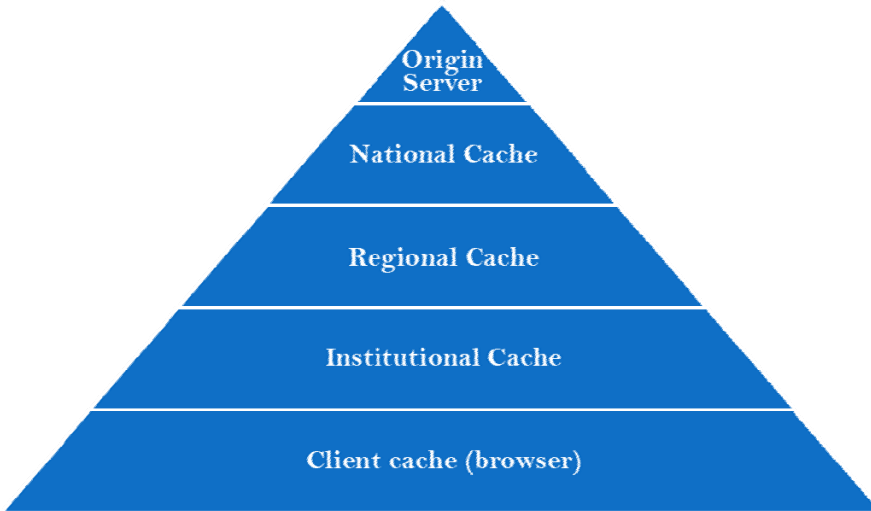


Fig. 1. Hierarchical Caching Architecture

2.2 Distributed Caching Architecture

Researchers have proposed an alternative to the hierarchical caching architecture and eliminated the intermediate tiers except for the institutional tier. All caches in that tier contain meta-data about the content of every other cache. Another approach proposed in this architecture is to employ the hierarchical distribution mechanism for more efficient and scalable distribution of meta-data (Wang, 1999). Figure 2 illustrates this approach.

In a distributed caching architecture that employs the hierarchical distribution mechanism, the layers that contain cached objects are only the client and institutional layers. Other layers contain information about the contents of the caches in the institutional layer.

2.3 Hybrid Caching Architecture

A hybrid scheme is any scheme that combines the benefits of both hierarchical and distributed caching architectures. Caches at the same level can cooperate together as well as with higher-level caches using the concept of distributed caching (Wang, 1999). A rough comparison between hierarchical, distributed, and hybrid caching architectures is shown in Table 1.

A hybrid caching architecture may include cooperation between the architecture's components at some level. Some researchers explored the area of cooperative web caches (proxies). Others studied the possibility of exploiting client caches and allowing them to share their cached data. One study addressed the neglect of a certain class of clients in researches done to improve Peer-to-Peer storage infrastructure for clients with high-bandwidth and low latency connectivity. It also examines a client-side technique to reduce the required bandwidth needed to retrieve files by users with low-bandwidth. Simulations

done by this research group has proved that this approach can reduce the read and write latency of files up to 80% compared to other techniques used by other systems. This technique has been implemented in the OceanStore prototype (Eaton et al., 2004).

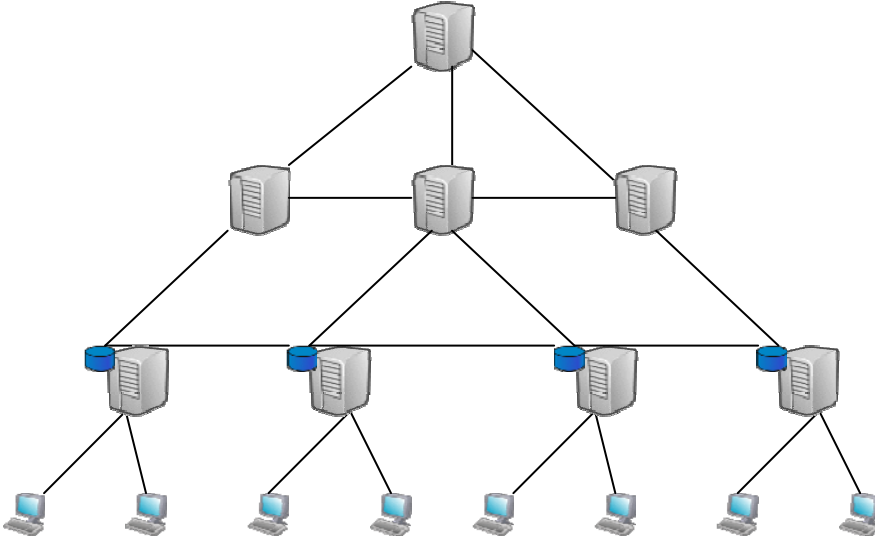


Fig. 2. Distributed Caching Architecture

Another study proposed an internet caching service called CRISP (Caching and Replication for Internet Service Performance). The problems that CRISP tried to solve are the performance, scalability, and organizational problems in large central proxy caches. The idea behind CRISP is to connect cooperative proxies through a mapping service which contains information about cached URLs in each proxy in the cooperative set. A drawback in this structure is the failure of the centralized mapping service. This drawback is solved by forcing the proxies to work individually without letting the user feel the impact of the failure except with the increase in the perceived latency. The study claims that this simple design and implementation of cooperative caching is effective and efficient for distributed web caches serving tens of thousands of users (Gadde & Robinovich, 1997).

Another study was motivated by the studies that have shown that limited bandwidth is the main contributor to the latency perceived by users. And also the fact that the majority of the population was still using modems at that time. An approach to reduce user-perceived latency in limited bandwidth environments is investigated. It explores a technique based on prefetching between caching proxies and client browsers. The idea is that the proxy would predict what the user might request/access next, where it invests the user's idle times while checking the result of the previous request and predicts what the user will request next. The proxy would push its prediction result to the client's browser cache, noting that the proxy will only use the contents of its cache in this prediction. The result of this investigation showed that prefetching between low-bandwidth clients and caching proxies combined with data compression can reduce perceived user latency by over 23% (Fan et al., 1999).

One analysis used a trace-based analysis and analytic modelling to put inter-proxy cooperation into the test and examine its performance in the large-scale WWW environment. It examines the improvement that such cooperation can provide in a 200 small organizations' proxies environments, as well as with two large organizations handling 20,000 and 60,000 clients. However the modeling considered a much larger population containing millions of users. The overall studies and examination done in this paper concluded that cooperative caching has performance benefits only within limited populations (Wolman & Voelker, 1999).

Another examination explored the benefits of the cooperation among proxies under different network configurations and user access patterns. This was achieved by classifying and analysing these cooperation schemes under a unified mathematical framework. These analytical results were validated using a trace-driven simulation. Using the results from the simulation and analysis, the following was concluded:

- Proxy cooperation is beneficial when the average object size is large and the working set does not fit in a single proxy. Such benefit also depends on the cluster configuration.
- The cooperation between proxies, where each proxy serves missed requests from other proxies in its cooperation set, is mostly sufficient when the users' interests are highly diverse.
- The cooperation of the proxies in object replacement decisions would result in more benefits when the user accesses are dense and the requests are focused on a small number of objects.

Overall, the benefit of the cooperation among proxies is dependent on a number of factors including user access, user interest, and network configuration (Lee et al., 2001).

Another study presented a decentralized, peer-to-peer web cache called Squirrel that uses a self-organizing routing substrate called Pastry (Rowstron & Peter Druschel, 2001) as its object location service. The key idea is to enable web browsers on desktop machines to share their local caches, to form an efficient and scalable web cache, without the need for dedicated hardware and the associated administrative cost. An evaluation of a decentralized web caching algorithm for Squirrel is also provided. Studies discovered that it exhibits performance comparable to a centralized web cache in terms of hit ratio, bandwidth usage and latency. It also achieves the benefits of decentralization, such as being scalable, self-organizing and resilient to node failures, while imposing low overhead on the participating nodes. Squirrel tested two different schemes called the home-store and directory schemes on a LAN organization. Performance studies have shown that the home-store scheme depicts less overhead on the serving nodes; this approach works for load balancing among the peer-to-peer nodes (Lyer et al., 2002).

Another research proposes a more effective use of caching to cope with the continuing growth of the internet. This proposal is to exploit client browser caches in cooperative proxy caching by constructing the client caches within each organization as a large peer-to-peer client cache. The goal of this study is to investigate the possibility and benefit of constructing such large peer-to-peer client cache in improving the performance of the internet. In this architecture, clients can share objects cached not only at any proxy in the cooperative set but also at any neighbour's cache connected to the network. After doing some simulations with/without exploiting client caches, results have shown that exploiting

client caches can improve performance significantly. It also introduces a hierarchical greedy dual replacement algorithm which provides cache coordination and utilizes client caches (Zhu & Hu, 2003).

Another study presented the design and implementation of a previously proposed scheme based on a novel Peer-to-Peer cooperative caching scheme. It considers new means of communication between cooperative caches. It also proposes and examines different routing protocols for data search, data cache, and replication of data. The results of the performance studies show the impact of cache coherency on the system's performance. It also shows that the proposed routing protocols significantly improve the performance of the cooperative caching system in terms of cache hit ratio, byte hit ratio, user request latency, as well as the number of exchanged messages between the caches in the cooperative set (Wang & Bhulawala, 2005).

Yet another study presented a trustable peer-to-peer web caching system, in which peers in the network share their web cache contents. To increase the trust-level of the system, they have proposed to use sampling technique to minimize the chance of distributing fake web file copies among the peers. They further introduce the concept of opinion to represent the trustworthiness of individual peer. A prototype has been built and the experimental results demonstrated that it has fast response time with low overhead, and can effectively identify and block malicious peers. This paper proposed a reasonable solution in locating the cached object using a search history similar to a log file that is stored in each peer. Each peer might carry a different log of search history of the peers in the system. When a request is initiated by a client and a cache miss was returned from its local cache, the request is forwarded to the client's nearest neighbour. If a cache miss occurred then this neighbour will look into the search history and find out which was the last peer that initiated a request to the same object and connect to that peer. This is an efficient solution but it would be rather faster if the client looks into the search history log it has before connecting to the neighbour in the first place. At the same time it can check if one of the peers that requested this object is any of its neighbours. Even though many papers have discussed the issue of trust between the peers, and some suggested "building trust" approach between peers, this issue is still largely unresolved and in need of further investigation (Liu et al., 2005).

3. Proposed Architecture

The proposed architecture is a cooperative client-client, client-proxy, proxy-proxy caching system that aims to achieve a broadband-like access to users with limited bandwidth. The proposed architecture is constructed from the caches of the connected clients as the base level, and a cooperative set of proxies on a higher level, as shown in Figure 3. The construction of the large client web cache is based upon some of the novel peer-to-peer (P2P) client web caching systems, where end-hosts in the network share their web cache contents.

3.1 Desired properties in the proposed architecture

The proposed architecture is based upon the idea of a hybrid scheme. It consists of two tiers of cooperative caches: client caches and proxy caches. The properties that we wanted to achieve while designing the architecture are as follows:

- Slight congestion in the parent caches.
- Low latency and data transmission time.
- Evenly distributed network traffic for faster transmission time and low latency achievement.
- Long connection times.
- Low bandwidth usage which is the priority in this architecture along with the low latency property.
- A maximum of two hierarchical levels.
- Low disk space usage therefore low duplication of objects.
- Maintain an easy plan to keep the cached objects fresh.
- Test different object retrieval approaches to achieve a high to a very high hit ratio and grant the user a fast response time.

Features	Hierarchical	Distributed	Hybrid
Parent caches	Congested	slight congestion	slight congestion
Latency	high	low	low
Connection times	short	long	long
Bandwidth required	low	high	low
No. of Hierarchies	<4	1	one - two
Transmission time	high	low	low
Network traffic	Unevenly distributed	Evenly distributed	Evenly distributed
Disk space usage	Significant	low	low
Placement of caches in strategic locations	vital	Not required	up to ISP
Freshness of cached contents	difficult	easy	easy
Hit ratio	High	Very high	high - very high
Response time	moderate	fast	fast
Duplication of objects	high	low	low

Table 1. Comparison between hierarchical and distributed caching architectures

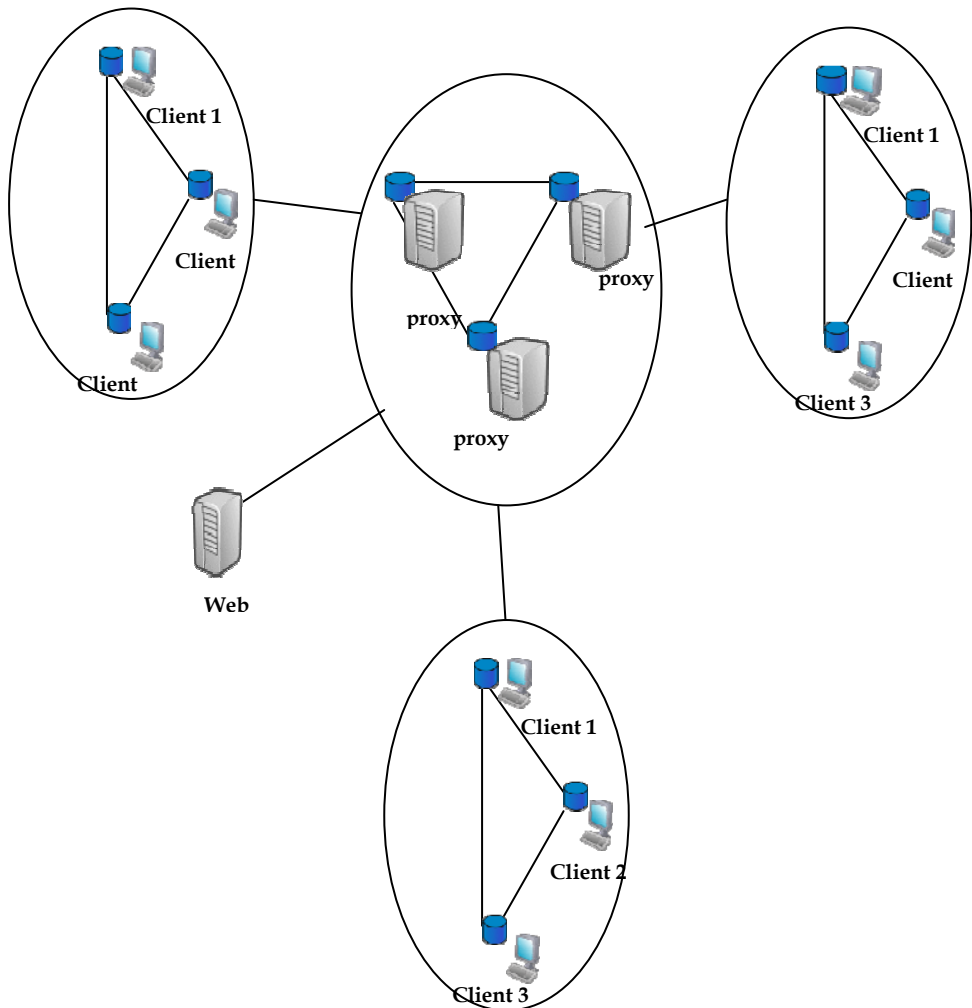


Fig. 3. The new proposed hybrid architecture

3.2 Considerations and design issues

There are many challenges in the proposed approach since we are dealing with an unknown number of clients in an unstable environment. We have chosen to deal with the following issues:

3.2.1 Cache Communication

The main challenge in cooperative cache architecture is how to quickly locate the location of the requested cached object.

Melpani et al. (Malpani et al., 1995) proposed a scheme where a group of caches function as one. When the user requests a page, the request is sent to some random cache. If the page was found in that cache, then it is returned to the user. Otherwise, the request is forwarded to all the caches in the scheme via IP multicast. If the page is cached nowhere, the request is forwarded to the home site of the page.

Harvest cache system (Chankhunthod et. al, 1996) uses a scheme where caches are arranged in a hierarchy and uses the Internet Cache Protocol (ICP) for cache routing (Wessels & Claffy, 1998). When a user requests a page, the request travels up the hierarchy to locate the cached copy without overloading the root caches by allowing the caches to consult their siblings in each level before allowing the request to travel up the hierarchy.

Adaptive Web Caching (Michel et al., 2001) builds different distribution trees for different servers to avoid overloading any root. This scheme is robust and self-configuring. It is more efficient with popular objects. For less popular objects, queries need to visit more caches, and each check requires a query to and responses from a group of machines. It is suggested to limit the number of caches the query visits, to decrease the added delay.

Provey and Harrison (Povey & Harrison, 1997) proposed a distributed caching approach to address the problems faced in the previously proposed hierarchical caching. They constructed a manually configured hierarchy that must be traversed by all requests. Their scheme is promising in the way that it reduces load on top-level caches by only keeping location pointers in the hierarchy (Wang, 1999). A simulation study was done as well, where the results showed that this proposed approach performs well for most network topologies. Results have also shown that in topologies where the number of servers in the upper levels is low, the performance of the hierarchical caching is better than the proposed approach. The conclusion of this paper is that the overall results show that there is no significant performance difference between the old and the proposed approach.

Wang (Wang, 1997) describes an initial plan in Cachemesh system to construct cache routing tables in caches. These tables guide each page or server to a secondary routing path if the local cache does not hold the document. A default route for some documents would help to keep table size reasonable (Wang, 1999).

Legedza and Guttag (Legedza & Guttag, 1998) offered to reduce the time needed to locate unpopular and uncached pages or documents by integrating the routing of queries with the network layer's datagram routing services (Wang, 1999).

3.2.2 Cache Coherency

The most outstanding benefit of web caching is that it offers the user lower access latency. It also defeats the side-effect of providing the user with stale pages (i.e. pages which are out of date with respect to their home site). The importance of keeping the cache's content coherent is to provide the user with fresh and up-to-date pages. Web caching reduces redundant data in the network which eases the process of keeping the pages updated. Some of the proposed mechanisms to keep cache coherency are strong cache consistency and weak cache consistency (Wang, 1999).

- Strong cache consistency
 - Client validation. This approach is also called polling-every-time. The proxy initially considers the cached pages are expired on each access and sends an If-Modified-Since header with each access of the resources.

- Server invalidation. When a resource is changed at the server, it sends invalidation messages to all clients that have recently accessed and cached the resource. The server has to keep a list of clients who requested and cached the changed resources which becomes unmanageable for the server when the number of the clients is large.
- Weak cache consistency
 - Adaptive TTL. The resource freshness problem is dealt with by adjusting the time-to-live parameter based on observations of its lifetime. If a file has not been modified for a long time, it tends to stay unchanged. Thus, the time-to-live attribute to a document is the current time minus the last modified time of the document.
 - Piggyback invalidation. Whenever a cache has to communicate with the server, it adds along with it a list of resources that are potentially out-of-date and asks for validation.

3.2.3 Cache Contents

Proxy caches have been recognized as an effective and efficient solution to improve the web performance. A proxy serves in different roles: data cache, connection cache, and computation cache. A recent study has shown that caching Web pages at proxy level reduces the user access latency 3% - 5% as compared to the no-proxy scheme (Wang, 1999).

It is very important to set the architecture and prepare it to deal with different types of resources. Most of the web resources are becoming dynamic with the invasion of web services. It is very helpful to use computation caching to retrieve dynamic data. It can be done by caching dynamic data at proxies and migrating a small piece of computation to proxies to generate or maintain the cached data. Also the architecture should be able to retrieve information about the requested resource before adding delay to the request by looking for it in the caches when it is an uncachable resource.

3.2.4 Load balancing

The hot spot problem is one of the issues that triggered the web caching research area. It occurs any time a large number of clients access data or get some services from a single server. If the server is not set to deal with such situations, clients will perceive a lot of delay and errors and the quality of service will be degraded. Several approaches to overcome the hot spots have been proposed. Most use some kind of replication strategy to store copies of hot pages/services throughout the Internet; this spreads the work of serving a hot page/service across several servers (Wang, 1999). Another approach that can be used is to get the server to work in a cooperative set with other servers or caches to fulfil a request without overwhelming the home server with users' requests.

3.3 Flow of information in the architecture

The flow of information in the architecture can have different scenarios and paths. The two scenarios chosen for this architecture are as follows.

Scenario1, each client keeps a search history log of the clients that contacted it. When a client initiates a request it first looks into its local cache. If the requested page is found, then it is fetched from the local cache of the client. Otherwise, it looks into its search history log

and search for the last client who requested this page. If found, it fetches the requested page from the client otherwise it consults the proxy to fetch the requested page. If the proxy finds it in its cache, it forwards the requested page to the client. Otherwise, it consults the proxies in its cooperative set. If none has it, then the request is forwarded to the home server (see Figure 4).

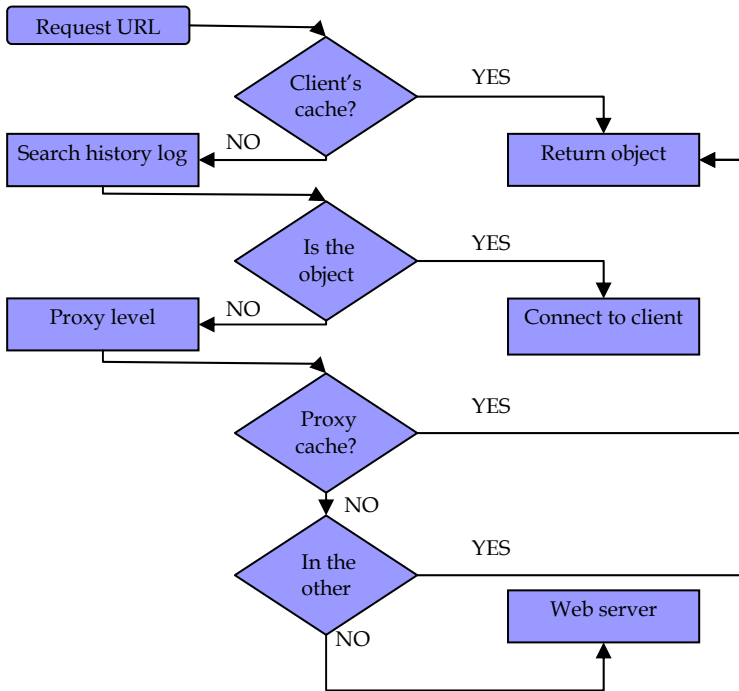


Fig. 4. Scenario 1

Scenario2, each proxy in the cooperative set is responsible of a group of clients that are geographically grouped. Also each proxy acts as the leader of the peer-to-peer connected clients and contains cache and routing information of the client caches. When a client initiates a request, it will first check in its local cache. If it is found then it will be fetched from the local cache of the client. Otherwise, it will send the proxy a page location request. If the page was cached in one of the client's caches, then it would forward the information of the client that has the page in its cache to the requesting client. Otherwise, the proxy will consult the proxies in its cooperative set and check if any of the proxies have the requested page in its cache. If none has the requested page, then it is fetched from the home server (see Figure 5).

Both of the mentioned scenarios are to be tested and analysed using a simulator. The simulation could result in the superiority of one of them or the need for a hybrid of both. The reason such scenarios are chosen is to explore and benefit from the free space offered by the client's caches when they are connected to the internet. This architecture aims to reduce

the load on the upper tier, the proxy, by initiating direct communication between the clients in P2P-like atmosphere which are geographically close to each other. The communication between the clients better stay as simple as possible as not to produce more delay and load on the client and organize the flow of the network traffic.

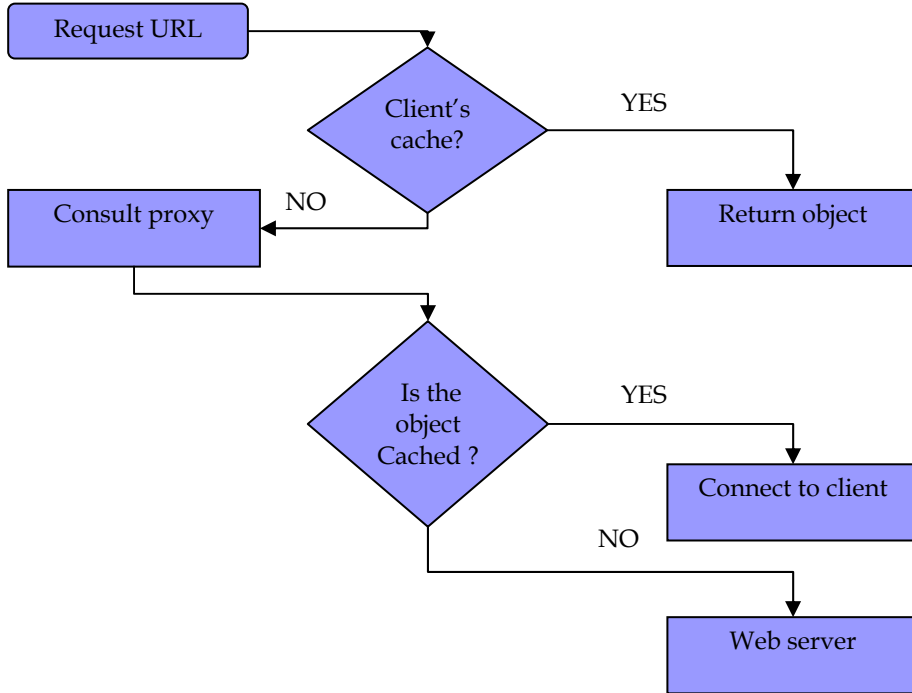


Fig. 5. Scenario 2

4. Conclusion

This chapter presented the basic web caching architectures that has been found in the literature as a solution for network congestion and server overloading problems. A rough comparison of common architectures has been presented to show the pros and cons of each. The chapter also proposed an architecture that is believed to offer a better performance in different aspects which is due to combining the benefits of many architectures and schemes into this new architecture. This architecture will look into some design issues such as the communication between the caches, the path to keep the caches' contents coherent, cache contents, and load balancing at the client and proxy side. Current work is on the simulation of the architecture's flow of information scenarios, using OMNET++, to obtain results and fine tune the architecture.

5. References

- Abrahams, M.; Standridge, C.R.; Abdulla G.; Williams S. & E.A. Fox(1995), "Caching Proxies: Limitations and potentials", in *Proceedings of the 4th International World-Wide Web Conference*, pp. 119-133, July 1995.
- Chankhunthod, A.; Danzig, P. B.; Neerdaels, C.; Schwartz, M. F. & Worrel, K. J.(1996), "A hierarchical Internet object cache", in *Proceedings of the 1996 Usenix Technical Conference*, pp. 13, 1996.
- Dykes, S.G. & Robbins, K.A.(2002), "Limitations and benefits of cooperative proxy caching", *IEEE Journal on Selected Areas in Communications*, Vol. 20, issue 7, September 2002, pp. 1290-1304.
- Eaton, P.; Ong, E. & Kubiawicz, J. (2004) , "Improving Bandwidth efficiency of peer-to-peer storage", in *Proceedings Fourth International Conference on Peer-to-Peer Computing*, pp. 80-90, ISBN 0-7695-2156-8 , August 2004.
- Fan Li; Cao, Pei; Lin, Wei & Jacobson, Q.(1999), "Web prefetching between low-bandwidth clients and proxies: potential and performance", *Performance Evaluation Review*, Vol. 27, issue 1, June 1999, pp. 178-187.
- Gadde, S.; Rabinovich, M. & Chase, J. S.(1997), "Reduce, reuse, recycle: An approach to building large internet caches", in *Proceedings of the Workshop on Hot Topics in Operating Systems*, pp. 93-98, ISBN 0-8186-7834-8, May 1997.
- Korupolu, M.R. & Dahlin, M. (1999), "Coordinated placement and replacement for large-scale distributed caches", in *Proceedings of the 1999 IEEE Workshop on Internet Applications*, pp. 62-71, August 1999.
- Lee, K.W.; Sahu S.; Amiri K. & Venkatramani C.(2001), "Understanding the potential benefits of cooperation among proxies: Taxonomy and analysis", *Technical report, IBM Research Report*, Septmber 2001.
- Legedza, U. & Guttag, J.(1998), "Using network-level support to improve cache routing", *Computer Networks and ISDN Systems*, Vol. 30, Issue 22 - 23, November 1998, pp. 2193-2201, ISSN 0169-7552.
- Liu, Jiangchuan; Chu, Xiaowen & Xu, Ke(2005), "On peer-to-peer client web cache sharing", *IEEE international conference on communications*, Vol. 1, May 2005, pp. 306 - 310.
- Lyer, S.; Rowstron, A. & Druschel, P.(2002), "Squirrel: a decentralized peer-to-peer web cache", in *Proceedings of the twenty-first annual symposium on Principles of distributed computing*, pp. 213 - 222, ISBN 1-58113-485-1, 2002.
- Malpani, R.; Lorch, J. & Berger, D.(1995), "Making World Wide Web caching servers cooperate", in *Proceedings of the 4th International WWW Conference*, Boston, MA, pp. 107 - 117, December 1995.
- Michel, S.; Nguyen, K.; Rosenstein, A.; Zhang, L.; Floyd, S. & Jacobson, V.(2001), "Adaptive web caching: towards a new global caching architecture", *IBM Research Report, RC22173*, September 2001.
- Povey, D. & Harrison, J.(1997), "A distributed Internet cache", in *Proceedings of the 20th Australian Computer Science Conference*, Sydney, Australia, February 1997.
- Rowstron, A. & Druschel, P.(2001), "Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems", in *Proceedings of the 18th IFIP/ACM International Conference on Distributed Systems Platforms (Middleware 2001)*, pp. 329 - 350, November 2001.

- Wang, Z.(1997), "Cache mesh: a distributed cache system for World Wide Web", in *Proceedings of the WCW'97*, Boulder, CO, June 1997.
- Wang, J.(1999), "A Survey of Web Caching Schemes for the Internet", *Computer Communication Review*, Vol. 29, issue 5, October 1999, pp. 36-46.
- Wang, J.Z. & Bhulawala, V.(2005), "Design and implementation of a P2P cooperative proxy cache system", in *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 508-514, September 2005.
- Wessels, D. & Claffy, K.(1998), "Internet cache protocol (IPC)", version2, RFC2186, 1998.
- Wolman, A.; Voelker, G.; Sharma, N.; Cardwell, N.; Karlin, A. & Levy, H.(1999), "On the scale and performance of cooperative Web proxy caching", in *Proceedings of the 17th ACM Symposium on Operating Systems Principles (SOSP'99)*, pp. 16-31, Kiawah Island Resort, SC, USA, December, 1999.
- Zenel, B. & Duchamp, D.(1995), "Intelligent communication filtering for limited bandwidth environments", in *Proceedings Fifth Workshop on Hot Topics in Operating Systems*, pp. 28-34, May 1995.
- Zhu, Y. & Hu, Y.(2003), "Exploiting client caches: an approach to building large Web caches", in *Proceedings 2003 International Conference on Parallel Processing*, pp. 419-426, ISBN 0-7695-2017-0, October 2003.

Heuristics of social process design

Gilbert Ahamer

*Institute for Geographic Information Science at the Austrian Academy of Sciences
Austria*

Abstract

The notion of design is prominent in the fields of heuristics, learning and gaming. "Design" can refer to space (architecture or geography), time (music) and even to roles and perspectives (e.g. negotiation games); such is named "substrate of design". The understanding of one substrate of design could be helpful for others regarding useful structures, evolutionary generation of such structures and applicable quality criteria.

"Life" (including gaming) is considered a continuous learning process on (i) personal and (ii) societal levels.

Rhythmisation, multi-perspectivism, and underdetermined gaming frameworks are identified as helpful structural principles and procedural values. These three help to provide recurring opportunities for gaming learners and other creative workers to "glue into the process".

It is proposed to discern four components of any social (gaming or learning) process and to notate them graphically in a manner resembling music scores, symbolized by the voices soprano, alto, tenor and bass. Such notation is applied to cases of simple and complex learning frameworks. Structural rhythms (e.g. such as "STAB" proposed here) are proposed to optimize complex societal learning procedures.

Keywords: design of gaming procedures; educational design; evolutionary societal procedures; game based learning; graphic notation; multi-perspectivism; quality criteria; rhythmisation.

Do you think a multi-disciplinary perspective could be helpful for finding solutions of new kind? Then try this text! It outlines a concept and a notation for social process design in gaming. I suggest writing down gaming procedures by means of musical scores.

1. Introduction

1.1 Objective of this paper: how to note down gaming processes?

The aim in the context of this paper is to optimize the design of learning processes (or more generally social processes) for individuals, groups of individuals and society as a whole.

The main concrete interest of this article is: how to note down social learning processes such as gaming? Such notation could eventually visualize helpful temporal, communicative and social structures in learning processes.

Hence, the issue at the core of this paper is the “*notation of social processes for gaming and learning*”.

The main question to be addressed and answered is: Which sequence of learning framework conditions delivers an optimal learning effect (understood here as change of behavior) independently of the initial stage of mastery of the learners?

1.2 Some definitions and key notions

In this text, learning is understood in multiple ways:

- learning = real change in behavior
- learning = creation of suitable and sustainable consensus
- life = a continuous learning endeavor
- living means learning; this creates reality.

- social process (sp) = element of SP (see Ahamer & Schrei, 2006: 226)
- societal procedure (SP) = compound of sp, can result in successful learning
- design = creation of helpful structures, can also be social design (= design of sp)
- mastery starts with “obeying rules”, and progresses to “creating rules” (Ahamer & Schrei, 2006: 235).

In this article, we see “design” in a generalized way as an array of temporal, spatial and interindividual structures inciting to enact procedures.

- games: the stage for enacting
- traditional learning: is mostly defined by conveying content
- unconventional learning: sets out to integrate learners into a suite of sp designed to incite them to maximize their changes of behavior.

1.3 Twofold relevance of “design of gaming and learning”

Hence, it is a valuable task to ponder *optimal structural design for such learning*, be it “simulation/gaming”, individual or societal learning.

Starting points are often attempts to “change the world” like interdisciplinary university courses, consulting in the public or municipal sphere, planning of sustainability measures such as climate protection or other complex learning endeavors.

The relevance of such questioning is only seemingly of purely pedagogical nature and relating “only” to learning individuals (micro-learning). Yet,

- (1) if life as such is understood as a learning procedure (i.e. changing previous behaviors) and
- (2) if “learners” are also entire societies and
- (3) if tasks exist also on a global level,

such learning endeavor may even pertain to optimization of global society’s answer to challenges such as global change (cf IPCC, 2007) or global warming (macro-learning).

To a certain extent, basic structural findings might be true for both orders of magnitude, namely for (i) individual and (ii) societal learning.

We therefore include the following examples:

(i) *Learning of individuals (defined here as micro-learning)*: Learning in academia or at university (e.g. interdisciplinary courses, needed in curricula like Environmental Systems Science (“Umweltsystemwissenschaften”: USW, 2007), or Environmental Management (Mayer et al., 2004, 2005; JDR, 2006; Lourdel et al., 2006; Zermeg, 2007)

(ii) *Learning of compounds of individuals (defined here as macro-learning)*: Distributed design (MacGregor, 2002), Computer Supported Co-operative Work CSCW (Heaton, 2002; Jarvenpaa & Leidner 1988; MacGregor, 2002; Wodehouse & Bradley, 2006; Johns & Shaw, 2006; Lloyd, 2004), societal learning: national climate protection targets (Crookall & Bradford, 2000; Kratena et al., 1998) that are not adequately met (WegCenter, 2007), low-energy building standards (HdZ, 2007; IP, 2006), sustainable urban renewal (van Bueren et al., 2006), municipal sustainability plans and energy concepts (KEK, 1997; LRP, 1995), approximation of Central European Countries to the EU (Ahamer, 2005), the “European Constitution” or a revolutionary new monetary system (Rauch & Strigl, 2006; Daly, 1999) and other examples of complex societal learning.

“Success” in learning is seen as effective change in real-life behavior, be it of (i) individuals or (ii) of societies. In section 4 it will be discussed, which four dimensions of human action should be affected by such change. Regarding the example of climate change, the stakeholders could be seen as facing severe difficulties in *really* changing (i.e. really learning) – as proven by ever increasing CO₂ emissions in some countries (UBA, 2004).

1.4 Why a “notation for learning and gaming processes”?

Development of a “notation for structures in social gaming processes” could help

- to optimize the usefulness and effectiveness of “social design” in learning frameworks when using graphical analogies (cf e.g. Hofmeyer et al., 2006: 432)
- to improve dramatic frameworks proposed for learning and societal procedures in general (cf Johns & Shaw, 2006, Roth et al., 2001)
- to visualize and hence conceptualize various levels of action and various modes of building of consciousness (cf Bilda et al., 2006)
- to depict the design of learning frameworks in a transparent manner.

What can be the role of “game based learning” (Prensky, 2001; Ahamer, 2004) for these targets? A short answer: A notation helps to better visualize the structure of learning games and negotiation games along time and with respect to expressible opinions. In order to safeguard high effectiveness of gaming, the “consumer” of such a designed “game” should be exposed to a sufficiently large number of changing surrounding conditions while flowing in the drain and train of the game.

In cases of suboptimal learning (e.g. in school or with climate change) a strong motivation arises to find means to improve learning procedures and strategies. Such is attempted here.

2. What can be designed at all? - Where can “design” emerge?

“The notion of design is prominent in the field of simulation/gaming. It has been a thread running through most work in the field.” (Crookall, 2003: 485) This section allows the reader to regard the theme “design” from a distance when focusing on “what might be designed”. Design pertains to various substrates (Table 1) thus resulting in diverse branches of design.

In this section 2, the following will be laid out: What is much needed is design (= structure in time, space, and human individuals) which facilitates the generation of the learning target, which is often the construction of a consensus - like molding an alloy of differing individual opinions.

Substrate of design	Branch	Example
<i>CLASSIC / already known traditionally:</i>		
(I) space	architecture	façades, buildings, arrays of rooms
(II) time	music	symphonies, polyphony of several voices
(III) geometry	graphic design	icons and logos
(IV) physical structures	painting, sculptures, fine art	forms and patterns
(V) mental structures	science	theories, "world formulae"
(VI) functionalities	engineering	machinery, industrial design
(VII) communication tools	distributed IT	www and CSCW
(VIII) acting humans	theatre on stage	threads of human action in a drama, choreography
<i>NEW / some are focused on in this paper:</i>		
(1) social processes	game play	free games, open learning (Ahamer, 2004; SiP 2004)
(2) individual interests	regulation by law	trial, jurisdiction
(3) effects of technologies on society and human environment	technology assessment = TA, environmental impact assessment = EIA	EIA procedures and laws (Crookall & Bradford, 2000; Aschemann, 2004)
(4) societal roles	Institutions, NGOs	politics, SGC (2006)
(5) rules for consensus building	Administration, diplomacy, UNO, ESD (2007)	spatial planning, Twinings in EU accession (EC, 2007)
(6) self-optimization processes	economics, politics	policy measures (Vester, 1980; Pilch et al., 1992)
(7) perspectives of a case	constitution and legislation	legal processes in a state
(8) creation of games and their rules	rules in a role-play, game based learning	gaming business, children play; i.e. "play with rules"
(9) evolutionary patterns	Genesis of transnational institutions: UNFCCC, EU; political game theory etc.	e.g. evolutionary economics, Global Change Data Base (GCDB, 2001)

Table 1. A very broad understanding of "design" would embrace a multitude of "substrates of design" in several branches. A designer is a "structurizer".

Here we understand “design” in a generalized way as structuring a range of substrates (such as the ones in the left column of Table 1) that could be effective to incite and enact procedures.

It might be a helpful objective to investigate whether a useful necessary design of structures in several fields resembles one another. Contemplating various disciplines (and their respective design structures) could aid in such overall endeavor. The following list and subsections outline first attempts.

In this sense, it could eventually be helpful to conclude

- from the inner deep structures (Casakin, 2004) of the classic substrates and applications of design (= upper half of Table 1)
- to the inner deep structures of the new substrates and applications of design (= lower half of Table 1) and to apply potentially new ideas of general relevance.

Some of the classically known substrates of design (upper half of Table 1) are briefly explained below.

- (I) Designing space: “architecture”, see chapter 4.2 and Fig. 4; examples: (Popov, 2002, Corbusier, 2007; Hofmeyer et al., 2006).
- (II) Designing time: “music”, see chapter 4.3 and Fig. 5; examples: (Beilharz, 2004, Xenakis, 2007; Zographos, 2007; SonEnvir, 2007).
- (III) Designing geometry: logos, see SGC logos in Fig. 3: convey condensed geometric information in a symbolic manner to denote mental structures; examples: (Schrei, 2006, 2007; Ahamer & Schrei, 2006: 239-242; Bouchard et al., 2006).
- (IV) Designing physical structures: arrays of color dots in impressionism (e.g. van Gogh), entire history of plastic arts.
- (V) Designing mental structures, e.g. theoretical mathematical concepts of mechanics:
 - (a) Aristotelian physics: force is proportional to speed in movement ($F \sim v$),
 - (b) Newtonian physics: force is proportional to acceleration ($F \sim a$),
 - (c) Modern quantum physics: a particle is a point in a space of possible states, concretely only determined after the act of measurement.
- (VI) Designing functionalities: “engineering”, e.g. of machinery: see journals such as *Engineering Design*. Functioning artifacts as products of “art”, e.g. cars, engines, industrial products; examples: (Bianconi et al., 2006; Ono, 2006; Boujut & Tiger, 2002; Badke-Schaub, 2004).
- (VII) Designing communication patterns: collaborating internationally and forming one product (CSCW, Wiki, e-learning: Herder et al., 2003; Ahamer, 2004), implemented from spatially and temporally distributed working places (Heaton, 2002).
- (VIII) Designing acting humans, i.e. theatre: interlinked threads of action and responsibilities: e.g. tragic situations, classic Greek drama.

Some of the newer substrates of design (lower half of Table 1) are discussed in the subsequent subchapters.

2.1 Designing social processes

What does it mean to design social processes (sp)? Here we define it to aggregate simple events (like learning content, peer review, debate etc.) as general elements of more complex long-term societal procedures. Such design was described earlier (Ahamer & Schrei, 2006: 226). Chapter 4.1 will give a practical example.

2.2 Designing individual interests

What does “design of interests” mean? Here it is understood as arranging individual stakeholders and the representation of partial interests in such a way that their intercommunication yields the best result possible, e.g. in a lawsuit or trial at court. It could be attempted to measure such results from the standpoint of the “common good”.

Design of interests can often be motivated as follows: if “fact-oriented” justice cannot be guaranteed sufficiently, at least a “best attainable solution”, namely “procedural justice” should be developed as a proxy to the best but principally unknown target of scientifically sound judgment in complex matters.

Legislation for “civil procedure” is an example here.

2.3 Designing TA and EIA

Technology Assessment means to weigh the desirable and non-desirable effects of new technologies on humans and the environment (Decker & Ladikas, 2004; Bechmann et al., 2007; Decker, 2001; Grunwald, 1999;) by means of a structurally designed “value benefit analysis” or “utility analysis” (Zangemeister, 1970).

In many countries, the legal implementation of the “culture of TA” is the “Environmental Impact Assessment” (EIA) for projects and the “Strategic Environmental Assessment” (SEA) for policies and measures (Aschemann, 2004; UBA, 2007).

2.4 Designing societal roles: “role design” and “institutional design”

What is a societal role? Roles are conceived here as seizable condensations of interests in the entire network and fluid of all theoretically possible interests, in other words the most relevant points in the “landscape of interests”.

“Design of roles” means to combine individual perspectives in a way that they form a potent and promising societal actor. For example, such analysis is exercised during coalition building for government or opposition in politics. One target is to reach a 360° panorama-like view of all possible perspectives; this facilitates consensus.

Designing societal roles and interests can lead to designing institutions. How is such “institutional design” performed? By establishing and founding concrete panels – such as for climate change, where IPCC (2007) was founded as scientific body and UNFCCC (2007) as administrative body – and clearly defining frequency and organization of their interaction and mutual responsibilities (e.g., UNFCCC each five years “commissions” an “Assessment Report” from IPCC which is then “acknowledged” by UNFCCC).

2.5 Designing rules for consensus building

What are rules? The borders restricting (potentially free) individual human action put forth in order to direct societal behavior in a desired way.

What is “rules design”? Making up one’s mind what to allow, restrict, enhance and discourage in terms of social action under guidance of a societal target (e.g. defining measures in economic policy).

It means to devise rules in a way that maximal blossoming of the positive potential of the actors is attained (e.g. students, pupils, and a country’s economy).

Principally, autopoietic development of rule design can also be hypothesized to occur as 4th generation of web based teaching (Ahamer & Rauch, 136), namely “to play with rules”.

Deep understanding of “design” (Casakin, 2004) would also incorporate steps to design social and institutional procedures such as: the EU programme “Twinning” (EU, 2005) enhancing “converging” of two formerly fundamentally different economic and political cultures to the “Copenhagen criteria” relevant for EU accession. Twinning helped all Central European candidate countries in acceding towards the EU by sharing the common body of legislation denominated as “acquis communautaire”, and designed as a hierarchical system of dialogues (Ahamer, 2005; SI, 2007).

In this respect, the distinction made by Heaton (2002) between rule-based cultures (like the Danish in their experiences) and personality-based cultures (like the Japanese) is helpful and a contribution to intercultural cooperation (Hofstede, 1994; GS, 2007).

2.6 Designing self-optimization processes

What are self-optimization processes? We understand them here as positive feedback loops that enhance the effectiveness of an initial action (e.g. of a political measure).

Design of self-optimization processes means to make use of systemic (economic, political, social) circular feedback mechanisms in order to reach self-sustaining policies that remain effective in the long run (Vester, 1980; Bossel, 1994; Pilch et al., 1992). Examples for (at least intended) self-optimization processes in macro-learning are global trading schemes for CO₂ emissions (ACCC, 2007).

An example for such design with respect to global change is: how to arrange tools and measures pertaining to reducing global CO₂ emissions: CO₂ trading, clean development mechanisms and other flexible instruments (IPCC, 2007).

2.7 Designing perspectives

What are perspectives? We understand them as outlooks onto reality that are partly pre-determined by the standpoint of the beholder.

How are perspectives (and their interplay) designed? Legislation sets out to manage and mediate between diverse views on everyday incidents (e.g. a traffic accident). Civil law, process law and administrative law and their contained procedural rules allot speaking right and time to parties in an individual judicial process or in a societal decision process. Such can be reproduced in a negotiation game (e.g., this is the sense of the gambling procedure in SGC level 3).

A practical example is the Austrian political system of “social partnership”.

2.8 Creating and designing role-play

What are roles? Based on the definition of perspectives above, roles are condensations in the patterns of perspectives, attached with the interest of persons.

Role-play (Corbeil, 2005; Prensky, 2001) can hence be designed in a way that gaming individuals develop a maximum of sovereignty and depth of action when striving for their interests. Many people slip in professional roles in the course of their lives and, in turn, are shaped by them.

Let us understand games as: The stage for such enacting of roles. Enacting means bringing to life. Giving it drama. Permitting ideas and perspectives to flow out of their containers and stream along the river bed of passing time. Logos (ancient Greek for "the word", "the idea") must be en-acted and incarnated. (equals also to: Ideas must be implemented in real life.) In extreme idealism (Moser & Moser, 2005) every individual's life is the physical manifestation of their mental values and consciousness.

Children during "free play" can often be observed to invent new rules, when they have "used up" the attractiveness of well-known games. – Both "homo ludens" and "deus ludens" (playing man and playing god) were conceived by Huizinga (1994).

2.9 Designing distributed structures leading to evolutionary patterns

What are "evolutionary patterns"? Here we try to understand them as way and path, along which our (techno-socio-economic and political) evolution flows, determined by its inner structure. Prevailing global evolutionary patterns could be understood and conceived as exponential (classic growth theory: Temple, 1999), stepwise (Raskin et al., 2002) or saturating (Daly, 1999). The crucial idea of system dynamics is that the inner structure of an interacting system (i.e. the architecture of its internal interconnectedness) determines the system's behavior along time (Ossimitz, 2000).

For example, when combining structural design and space design, the important idea is "to include with the layout of space also the layout of basic structure." (Hofmeyer et al. 2006: 434). Each piece of architecture leads to the evolution of typical patterns of social behavior inside it, including a typical pattern of rules of behavior (e.g. kitchen rules in a hostel).

In a systems analytic approach, the generation of rules in a (social) system in itself is seen as a result of the system's state (Ossimitz, 2000). Adding an evolutionary approach, consecutive phases of system growth are producing different sets of rules along consecutive phases (compare chapter 4.5 or Ahamer, 2003: 8).

Examples could be: developmental policy, namely how to promote a country's autonomous growth and ability to help itself (GS, 2007).

To summarize chapter 2, a "designer" can be a *structurizer regarding a large variety of substrates*. Consequently, a designer's outlook is predisposed to reach beyond the contingencies of the prevailing substrate and to touch down to the patterns and principles construing the structures.

3. Three theoretical principles for the “design of social processes”

The task of a suitable learning framework (be it micro or macro) is to design along time suitable social, gaming and other structures that are intended to change real human behavior. This chapter proposes three principles that may aid in this designing task:

- Rhythmisation (3.1)
- Multi-perspectivism (3.2) expressed through roles
- Underdeterminism (3.3).

3.1 The value of rhythmisation

Rhythmisation offers an ever changing structure to the eye, ear or spirit of the individual who consequently is more easily able to “glue into”, integrate or resonate with the offered structure. For example: a lecturer inserting short stories into a long explanation recaptures attention of students with more practically oriented learning profiles.

Rhythmisation as a theatrical means for structuring processes in time with changing speed of oscillation for the dramatic interaction of actors allows for various characters of spectators. The intrinsic time constant of each individual to act, react and allocate interest will be met with higher probability.

With a façade, rhythmisation allows the eye to better discern a largely perceived horizontal area, to pick up the pieces of the façade and to grasp the offered structure more conveniently, e.g. aided by baroque risalits or pilasters. The eye (symbolic for “pre-understanding”) retrieves more easily a subsequence which addresses the spirit of the onlooker because it comes into resonance with similar mental predispositions.

In didactics, rhythmisation (of the elements of actions, “sp”) is a necessary structure in order to provide recurring opportunities for learners and for other creative workers to “glue into reality” – according to the understanding „double interact“ (as Weick (1979), or Klabbers (2003: 577) have put it.

3.2 The value of multi-perspectivism and its expression through roles

Multi-perspectivism means to be able to adopt and understand another standpoint. Multi-perspectivism is a crucial step towards the ability of reaching consensus and a means for locating, organizing and measuring perspectives while consciously abstracting from one’s own position. Multi-perspectivism may be incited and aided by spatial and temporal segregation of views (e.g. each team is sitting at separate tables) which causes repositioning of perspectives.

For this paper, roles are facilitators for adopting different perspectives.

One possible application are resulting IT tools (e.g. CSCW: Heaton, 2002, MacGregor, 2002) where cultural preconditions shape the type of social processes occurring among stakeholders (see chapter 2.2). Multi-perspectivism is a key structure that learning tools/offers should provide in order to be helpful in a pluralistic society.

What are perspectives and roles in general? When reverting the direction of reasoning in chapters 2.4 and 2.7, they can be understood as enforced particularizations of the entire view (the contemplation of the whole), which are caused by our earthly restrictions of space-time structure. In Moser & Moser’s opinion (2005: 221), on the ethical, humane or ontological levels, such restrictions can only be surpassed by forgiveness between human individuals,

who – despite their individuality – are seen as (singularized, i.e. colored) reflections and facets of (holistic) white sun light, just like in a prism.

In a similar understanding, game play is the intentional (i.e. for pedagogic reasons) demolition and fragmentation of a holistic world view into the facets of the single roles' perspectives. Such destruction and subsequent reintegration of facets into a whole is trained in negotiation games (Dong, 2007). It could be said that negotiation games deliberately deconstruct the dimension of "opinions and world views" and artificially map it into the dimension of "time during game play" or other mapping or framing.

3.3 The value of underdeterminism in game-play

Underdeterminism means that a system offers more than one degree of freedom (in physics) for the motion of a particle or (in games or real life) for decisions of an individual. As said above, roles in game play are the playable reification of such (underdetermined, hence) different worldviews and world perspectives.

Thus, it is appropriate to "oscillate" or "dance" between these world perspectives (e.g. using roles) in order to arrive at a consistent 360° panoramic view. Such "trial and learning" based motion is allowed (only) in underdetermined systems (see Fig. 1). Therefore, if suitably arranged (by a system of loose but enabling rules), underdeterminism can enhance learning (hypothesized again for both micro and macro levels).

A symbolic physical example for the value and appropriateness of underdeterminism in real life can be the bicycle rider who oscillates around an "ideal path" of trajectory when continuously correcting their body's lateral inclination through steering their handlebars.

Games can be structurally understood as a shimmering of situations, as an unstable balance with shallow local optima (mathematically speaking). Compare a football game: the direction of the game continuously changes, one instable state is followed by the other, and predictability is almost non-existent. As a result, the dynamic situation permanently balances on a knife's edge.

Continuous change of standpoints, viewpoints, perspectives, and strategic constellations occurs frequently. Football players break clear, change boundary conditions for others and open avenues for new tactic actions. By acting and running, players create the game plan for others. According to design literature, iterative oscillation occurs between the problem space and the solution space (Maher, 2000; Dorst & Cross, 2001: 434).

3.4 Both design and gaming need underdeterminism

Underdeterminism is characteristic for design as such (Restrepo & Christiaans, 2004) and needed for game based learning. Therefore, learners are best provided only a loose corset.

In gaming, how loose should structures be? Societal procedures need space and liberty to grow properly and fruitfully. "Games" seem to be a promising environment to allow for such liberty in complex human action and can be called a "stage": "Simulation games provide a safe, condensed and dynamic environment, based on reality, in which participants, either professionals or students, can experiment with decisions and negotiations" (Mayer and Veeneman, 2002).

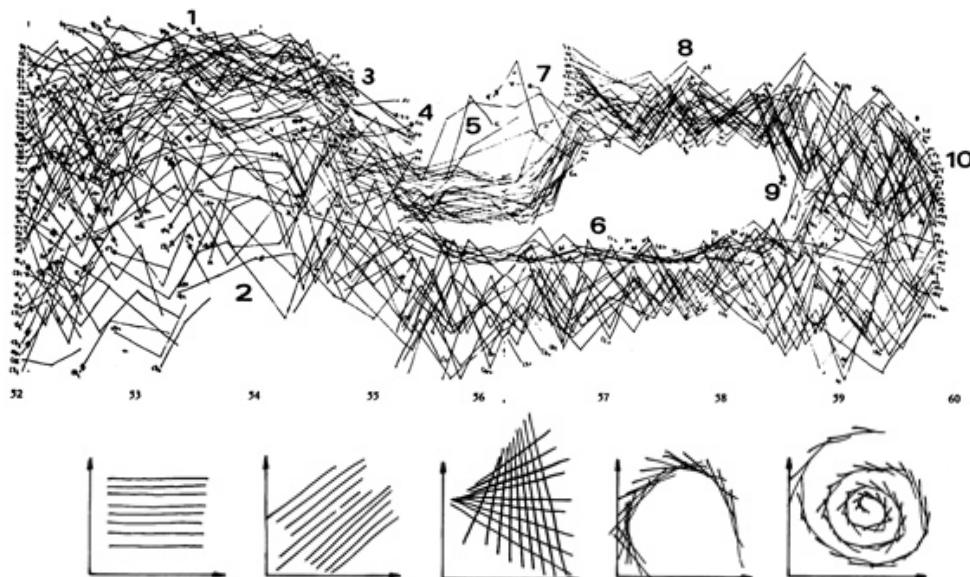


Fig. 1. Intentionally organizing “underdeterminism” in music (by Xenakis) means “playing”. Source: (Zographos, 2007); Figure 11 in (Beilharz, 2004).

The stroke of the painter or violinist (Fig. 1) is essential for art and characteristic to the artist. For a painter, it may mean to intentionally decrease optical resolution or preciseness in order to allow for another reality to enter, additionally to direct optical and imminent physical reality.

“Freedom of an artist” is the deliberate deviation from what is considered “real”. In didactics, this translates to more or less deliberate deviation from a standardized learning path. The innovative learner cannot always be kept on track on the ideal learning path (ideal only to a traditional designer). Such is another expression of underdeterminism.

Any notation in art is loose enough to allow space for interpretation of the performer.

Concluding chapter 3 and in the light of the above, facilitating (working) life (one of the aims of designers, see Heaton, 2002) – i.e. facilitating learning according to chapter 1.2 – can therefore be achieved by suitably structuring dynamic processes of enacting individuals’ opinions.

4. How to note down multidimensional rhythms

4.1 Dramatic Rhythm: a negotiation game

The sense of “noting down” a rhythm is to distil out of it the crucial structures. As a first concrete example of rhythmisation in social interaction, an attempt to note down the dramatic procedure of “SURFING GLOBAL CHANGE” (© Gilbert Ahamer) is shown in Fig. 2. This negotiation game is explained in (SGC, 2006) and is taken here as an example because its activities (Fig. 3, central column) contain all three of the above principles.

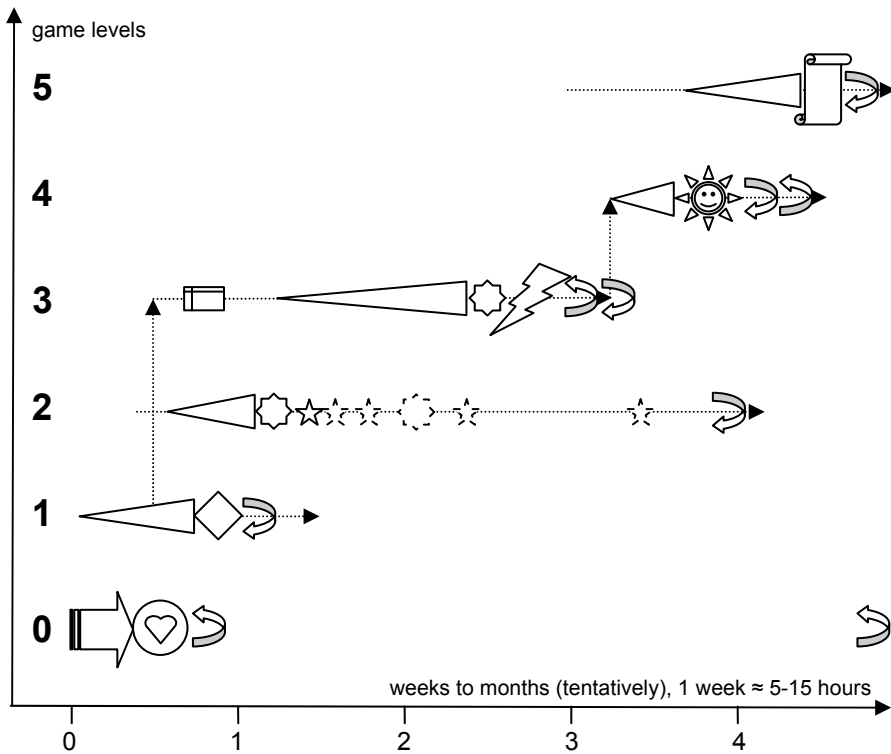


Fig. 2. Graphic representation of activities and time dynamics of the five levels in SGC (2003 style of SGC graphics).

Legend:

- Solid symbols vs. dashed symbols = compulsory vs. optional activities
- dotted lines = participants are informed about the phase
- triangles with growing thickness = preparation phases
- looped arrows up/down = feedback to facilitator/to participants for debriefing
- diamond = quiz in Level 1
- stars = declarations of points of view/reviews/updates in Level 2
- matrix = convene on two themes & develop two discussion matrices for Level 3
- flash = confrontational discussion in Level 3
- sun = consensus oriented discussion in Level 4
- document = integrative interpretation of global trends as 360° view in Level 5.

While using the numerous tools of web-supported learning, the online functionalities shown as logos in Fig. 3 are used for the rhythmized activities of the students.

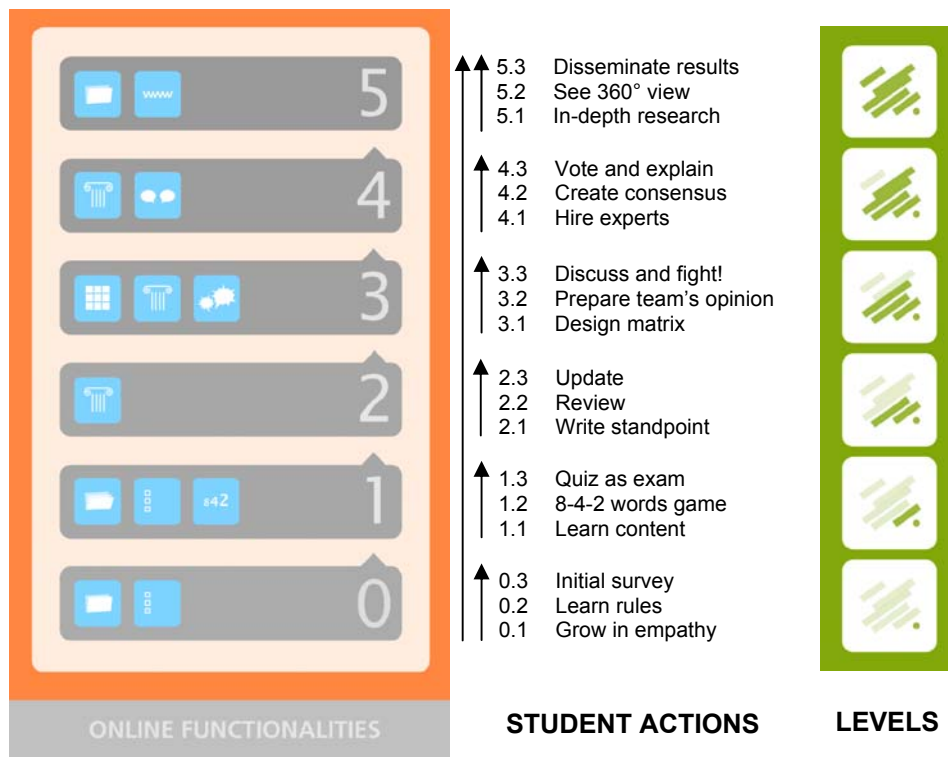


Fig. 3. Online functionalities and student actions in the five levels of the negotiation game SGC (2005 style of graphics by C. Schrei, cf Fig. 4 in Ahamer & Schrei, 2006).

4.2 Rhythm in space: a façade

After viewing the dramaturgy of “social processes”, the rhythmisation of the façade of the *Couvent de la Tourette* by Corbusier and Xenakis (see lower image in of Fig. 4) could be regarded as structurally similar (in the sense of Table 1) to the intended rhythmisation of the different communicative procedures in SGC (Figure 6). Both, in fact, resemble a musical score (Fig. 4 above left, compare with center and right).

Rhythms in several “storeys” vary independently from one another, shown by varying density of vertical window pane delineations (Fig. 4 above centre).

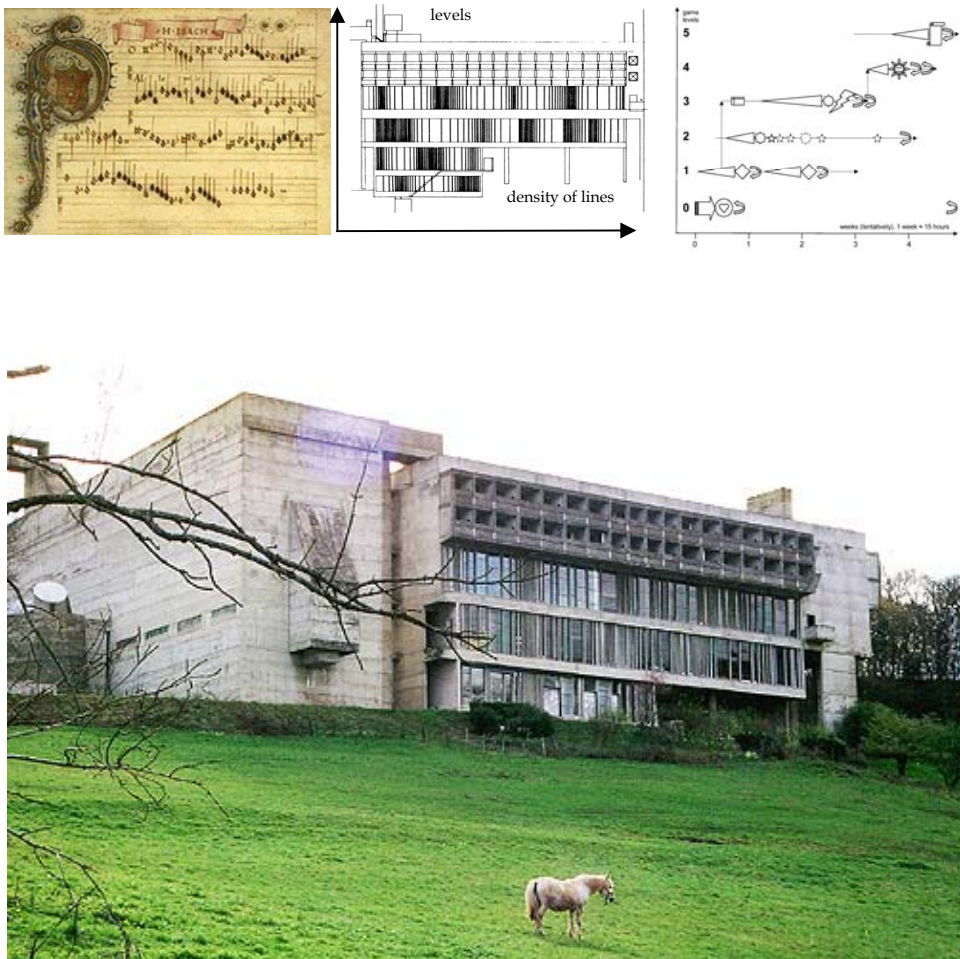


Fig. 4. Rhythmisation in space (above centre, façade by Xenakis: photo below) and in time (above left: music; right: symbolic notation for SGC's five levels dramaturgy: above right). Image sources, starting from left: (Vocal Consort, 2007), Figure 16 in (Beilharz, 2004), (Couvent de la Tourette, 2006).

4.3 Rhythm in time: music scores and flow charts

The historical and a modern type of musical notation are shown in Fig. 5 (left and right). Most often, different human voices are represented by different layers or levels in the notation.



Fig. 5. Musical scores: rhythmisation of polyphony in time (left: 18th century; right: 20th century).

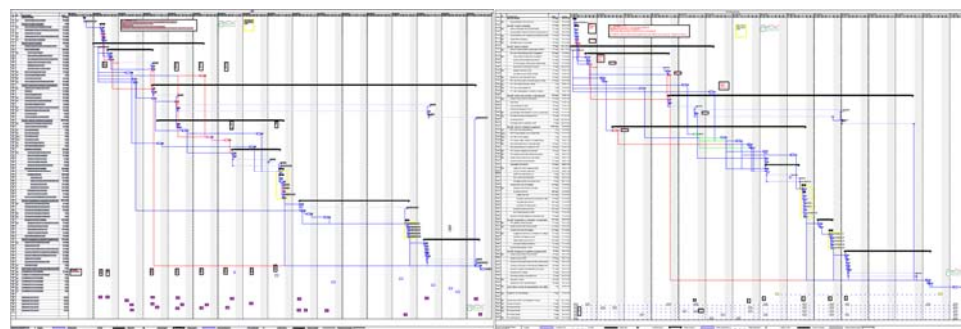


Fig. 6. The sequence and structure of two SGC implementations at Graz University in 2003-2005 as mirrored by the software for project organisation “MS Project”. Each course lasted one semester and included 3 to 5 lecturers.

The composition of a project plan of SGC (Fig. 6) comprises several lecturers from various disciplines whose contributions are blended into each other. Its organisational structure resembles the above notations.

4.4 Structural similarities between music and gaming

When en-acting (e.g. a composition of music), the temporal structure follows a detailed poly-rhythmic sequence – just as in Xenakis’ façade (Fig. 4). If adding the comparison between singers’ voices (soprano, alto, tenor, bass) and the stakeholders’ perspectives on the issues (e.g. using the roles industry, ecologists, administration, citizens), namely “polyphony”, it could be hypothesized that a suite of games such as SGC resembles a structured choir with changing roles of carrying and varying the musical motives and melodies.

As can easily be seen from baroque scores, music has progressed to polyphony already quite early in history. Such might serve as symbol for “orchestrated polyphony of views”.

Music is characterized by strict enforcement of coherence of the single voices. This hints to the importance of coherence (or at least good timing) of the different “social voices” in a societal learning procedure.

4.5 Structural similarities with evolution

Similar to the graphic impression of Fig. 1, the dynamics of global techno-socio-economic evolution show highly underdetermined behavior. Fig. 7 above shows the average development paths of the annual growth rate of energy consumption, plotted against a proxy for “economic stage of development”, namely per capita economic activity (GDP/cap), for all countries in the last 30 years (Ahamer, 2003) (1 red line = 1 country).

The more countries develop, the closer they seem to gather around a path that leads from slightly positive to slightly negative energy growth rates - the latter projection would actually help climate protection. Evolution “heads for different targets” during consecutive phases, such interpretation might be possible (see chapter 2.9). Initially, civilizations develop ever increasing hunger for energy, which saturates later on.

A different graphic impression confirms Fig. 7 below, which for each continent plots vertically “the level of mechanization in agriculture” versus horizontally “land needed per unit harvested”. Lines are moving leftward while mainly affected by annual weather changes that affect harvest. Again, the *target of development depends on the phase*.

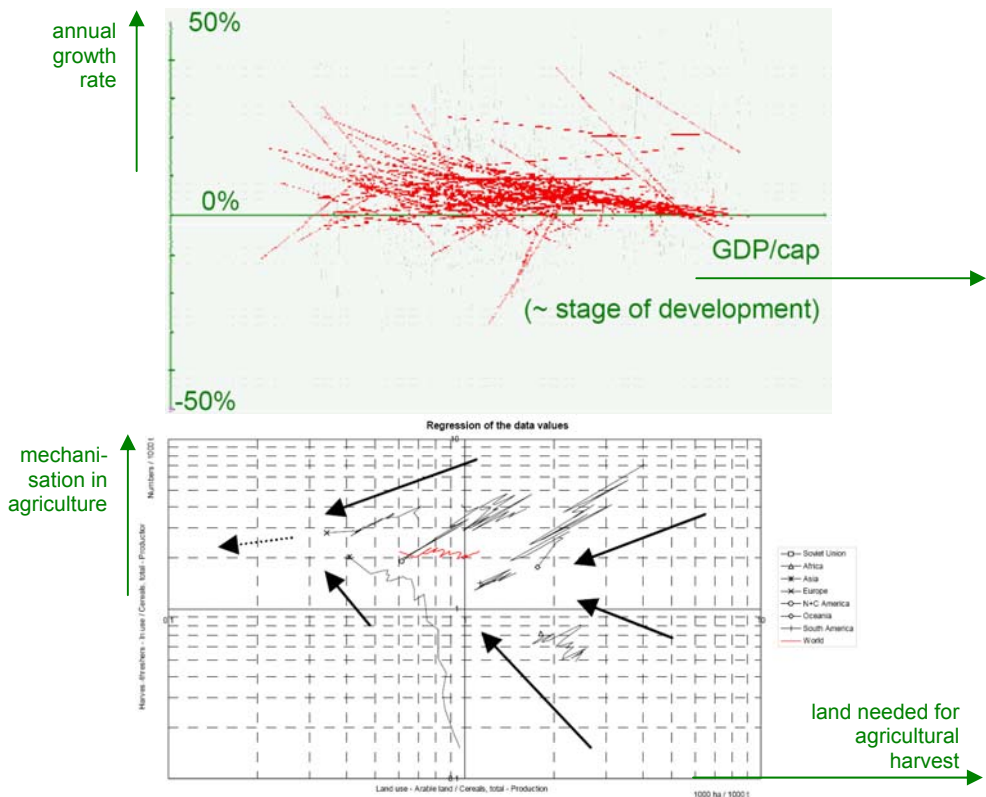


Fig. 7. Above: per country growth rate of energy demand plotted against per capita economic activity (GDP/cap), this growth rate apparently decreases along evolution (Ahamer, 2003). Below: per continent “machinery per crops harvested” versus “arable land needed per crops harvested”, these evolutive paths could be continued as dotted.

Note: in the next chapter, the structural similarity between ‘music’ and ‘societal learning procedures’ does not mean “a voice equals a stakeholder” but pertains to structural characteristics of social procedures. This hint is given for better understanding of the following type of notation!

5. The notation of social processes

In this chapter, a new notation for social processes (elements sp) will be proposed which resembles musical scores. The notes in music represent elementary “social processes”, the melody an entire “societal procedure” like learning, designing or even protecting global climate.

The resulting “social scores” sketch the communicational structure along time in various communicative dimensions.

For this paper, a new and original type of notation is developed.

5.1 Which basic dimensions exist in social processes?

When designing societal procedures, the intensity of social characteristics may vary along time. Once the emphasis is placed on teamwork, sometimes on individual work, here on understanding, there on confrontation. Can we invent a set of very basic and fundamental “dimensions” prevailing in any societal (gaming or learning) procedure? This chapter tries to do so.

Remembering one’s own experience in singing, the reader might figure out that often soprano carries the melody and the reader may enjoy looking for other structural similarities. Table 2 attempts to introduce an order of fundamental social dimensions which in this paper will be symbolically called “voices”.

Inspired by the examples in chapter 4 an attempt is made to split up the different components of human behavior into distinct levels in a graphic representation.

These four different components of human action are discerned based on practical experience with SGC for several years. They are thought to be organically independent (mathematically speaking: linearly independent) characteristics and components of overall social and societal behavior, be it learning, working, teaching or even politics (Table 2).

voice in music	structural functionality	gaming in education
S = Soprano	Leads the melody	Logical <i>information</i> conveyed
A = Alto	Follows the melody	<i>Team</i> building
T = Tenor	Counterfigures to melody	Debate & discourse, <i>dialogue</i> of facts
B = Bass	Longstanding cord basis	<i>Integration</i> with others’ experience

Table 2. Suggestion for structural similarities between music (left) and gaming in education (right).

In Fig. 8 this systematization is depicted, including arrows for a first orientation.

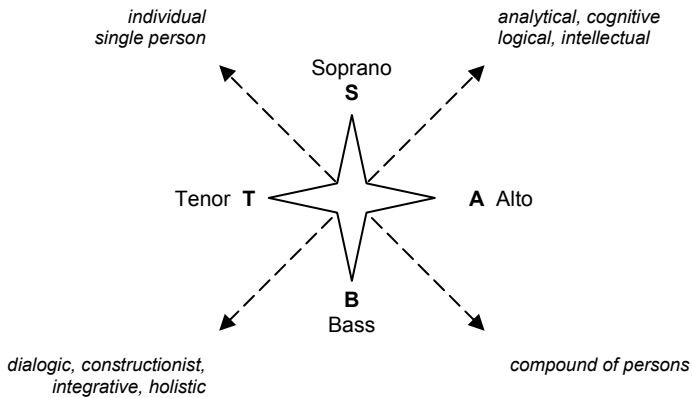


Fig. 8. The meaning of the “four voices” described in Table 2 as basic dimensions for any social process in a basic communicational structure; resembling a wind rose, including dashed auxiliary characteristics valid for two neighboring dimensions.

This sketch might resemble a wind rose as known from classical geographical maps. The meaning of the four voices (= four ends of the wind rose in Fig. 8) is allocated to basic properties of practical human life (dashed arrows) in a pairwise manner, in order to facilitate systematization.

Often, there “is only one soprano voice” in traditional pedagogy (or classical climate policy), if it restricts itself to “only conveying information”. Quite on the contrary, the conviction of the present text is that the four fundamental “dimensions” in Table 2 span up a four-dimensional vector space of action (as in vector calculus, Fig. 9), in which a rich multitude of “societal procedures” can occur. Such multitude might be very helpful – or even prove indispensable for success, if only adequately designed.

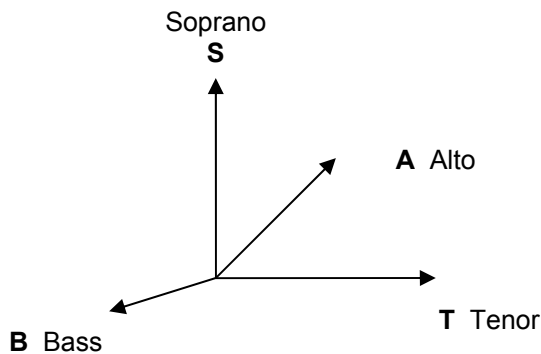


Fig. 9. The four voices span up a four-dimensional vector space of description in analogy to a vector in 4 dimensions.

In order to reflect better the new understanding and to systematize Table 2, the following interpretation of its right column is suggested (see also the logos in Fig. 11 right):

- S = starting point in the *world of facts*: information, “content”, “truths”
- A = starting point in the *world of relationships*: communication between individuals
- T = deal with S = “handle the arrays of facts and information”
- B = deal with A = “handle the arrays of communicating individuals”

In this possibly new and mathematically formalized understanding, each societal didactic incident equals one point in this four-dimensional space which is defined by values of (info; team; debate; integration), just like a vector in a “state space” in modern physics.

It is possible that the attention of music listeners fluctuates from one voice to another as the guiding motive might be repeated or reflected by other voices (as often in classical music). Similarly, the main attention of a game player could switch from one social functionality to another during educational game play.

Not a synchronization (as light waves in a laser beam) of learning opportunities for all students in the entire class is chosen for SGC (e.g. one lecture for all, then one exam for all), but rather a variety of situations with combined opportunities (e.g. team formation with underlying monitoring of academic skills within the team based on a preliminary literature research). However, such variety needs to be suitably structured, which is exactly the ultimate target of “social process design”: firstly for educational purposes, but secondly also on the societal level for appropriately reacting to climate change, global change and globalization.

Regarding didactic theories (that might span from cognitively oriented to libertarian), we need a “do the one and don’t let go of the other”! The same applies to climate change.

After having outlined general principles of “game design” in education, the next section goes into details of a rhythmized structure using SGC as an example, after starting with the simplest settings.

5.2 Notation for simple learning environments

The following scores deliver simple examples of social procedures intended for learning. Quantification of the four dimensions is based on practical experience.

Example 01: Classical teaching (left in Fig. 10) is often concentrated on conveying and imposing content “S” (of often rather classical nature) on students and shows no activity in the other dimensions (T, A, B). Consequently, team building among students is not observed, and no critical discussion of basic assumptions or subsequent valuation of contradicting scientific world views takes place. Similarly, classic administration of climate protection might restrict itself to discussions inside a small community of professional stakeholders with little emphasis on team building or even interplay with newly forming NGO’s. Climate policy purely administered “via decree” runs into the ever existing bottlenecks of harsh reduction in budgets and might ultimately remain unsuccessful as a result of such restrictions.

Example 02: Contemplative learning: Another monodimensional approach to learning based purely on “sensitively exploring the self” (right in Fig. 10) and constructing an image

of the world from the source of one’s own imagination is “learning from one’s own contemplative experiences”. Such an approach lacks connecting personal impressions to academic mechanisms of iterative critique, as well as reconsidering particular views of one’s own small community. “Contemplative climate policy”, as a hypothetic example, would remain restricted to the arbitrary mists of wishful thinking on the basis of a “oneself knows best” attitude inside self-reassuring particulate stakeholder groups.

Other one-dimensional approaches could be hypothesized: only “A” as ‘team’ dimension as in workplaces with many specialized teams that are not mutually interacting (Wodehouse & Bradley, 2006); only “T” as ‘debate’ dimension that might suffer from too shallow academic level and from low centripetal cohesion of discussants. Generally, it can be understood that exaggerating one dimension at the expense of the others does not tend towards decisive success.

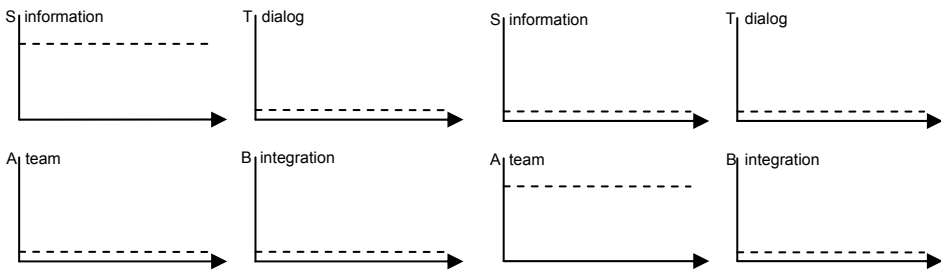


Fig. 10. Notation for simple learning environments (left: example 01, classical teaching; right: example 02, self-centred contemplation).

Example 03: Simple role-playing: Initial package of content followed by social learning with subsequent role-playing, e.g. in the style of systemic interventions (Hellinger, 2000), socio-drama (Weinberg, 2007), or short and simple simulation games (Fig. 11 left). The onefold switch from S to B generates some, but only moderate levels in A and T.

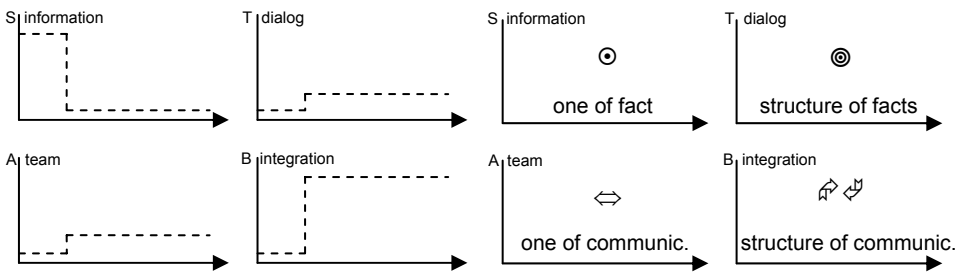


Fig. 11. Notation for simple role-playing (left, example 03). A graphic legend to this series of scores (right) recalls the structure of thought by means of logos.

The legend in Fig. 11 (right) classifies the four dimensions (as mentioned above in 5.1 and defined in Table 2) and uses simple logos for easy orientation in the diagrams.

5.3 Notation for a simple structured learning environment

Example 04 for a simple structured learning procedure: the $3 \times 7 = 21$ lecture type (Ahamer & Rauch, 2006; Akademie 2006):

Several "Interdisciplinary Practicals" (IP) of the curriculum on "Environmental Systems Analysis" (USW, 2007) – a worldwide unique and systems oriented type of master studies at Graz University – have been organized by interdisciplinary teams of students and lecturers (IP, 2005) for groups of 21 students for complex themes (e.g. Global Change) along the following procedural shell while using support from web based learning platforms (listed incl. abbreviations used in Fig. 12):

1. Each lecturer conveys initial information in introductory blocks ("Info 1")
2. Each student authors an individual seminar work ("Individ.")
3. Each lecturer deepens information in consolidation blocks ("Info 2")
4. Seven groups of three students (of the same scientific discipline) collaborate and create a common standpoint on their sector of the theme ("Intradisc.")
5. Three groups of seven students (of different scientific disciplines) collaborate and create a common standpoint on the entire theme ("Interdisc.").

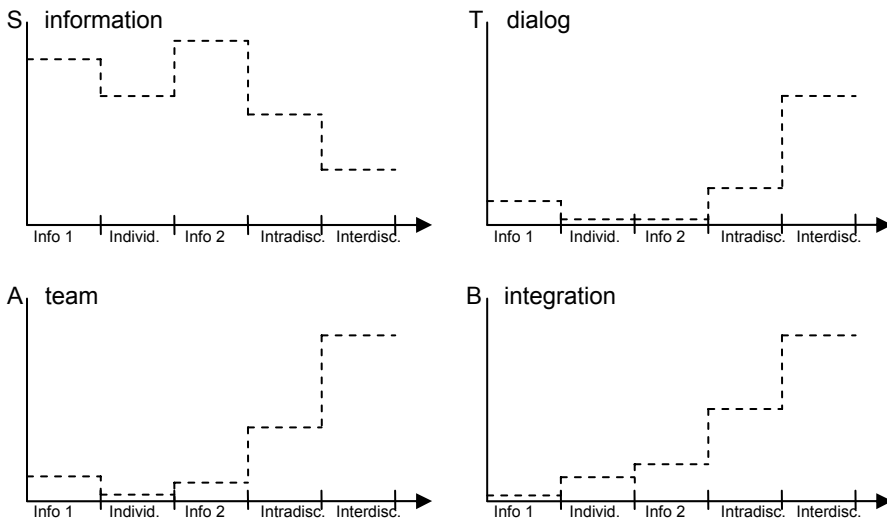


Fig. 12. Notation for a simple structured learning environment: "3x7 = 21" (example 04).

As can be seen from the figures, it is the aim of the structural "design of social learning process" to reach (and maintain, if possible) during the entire duration sufficiently high levels in all four dimensions. According to the above graphs it seems possible to hypothesize that *a change of state in one dimension creates a shift in another dimension*. Consequently, well-deliberated changes in "framing situations" applied by the moderator (or dramatic designer, e.g. a lecturer) can lead to increased values of other dimensions (which – in a largely unknown way – are apparently complexly related to each other). This preliminary hypothesis will yield a conceptual model in chapter 5.6.

5.4 Notation for a complex suite of game based learning

Example 05: the complex negotiation game: "SURFING GLOBAL CHANGE" (SGC, 2006; Ahamer, 2004).

The *design of social game dynamics* is key to "SURFING GLOBAL CHANGE" as explained earlier (Ahamer, 2006). SGC falls into several categories of "International Relations" gaming (Crookall, 2003a: 221). It uses web based media for social innovation. The main interest of SGC is not so much a technologically highly sophisticated product but the design, interplay and timing of the involved social procedures (briefly explained in chapter 4.1 and Fig. 3).

Building on chapter 3.2, a crucial idea of SGC is that the change of roles is repeated (from actor to spectator and back), which trains wandering and oscillating perspectives and keeps the muscles of the eye trained to adapt to various perspectives (Kristjánsson, 2006: 52):

- In level 1: from expressing to selecting coined short explanations
- In level 2: from reviewer to being reviewed
- In level 3: from one stakeholder to the other
- In level 4: from discussing to being monitored
- Level 5 sets out to integrate these perspectives in a composed "360° view" like a baroque world theatre.

Therefore, the macrostructure and microstructure scores of SGC appear as follows, based on experience with a dozen game implementations to date:

Example 05a: macrostructure of scores of "SURFING GLOBAL CHANGE":

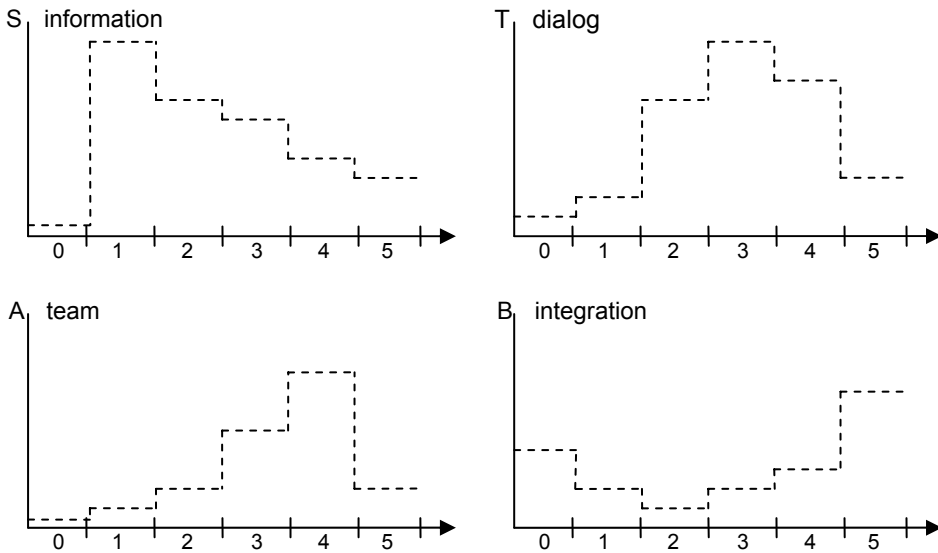


Fig. 13. Notation for a complex suite of game based learning: "SURFING GLOBAL CHANGE" - Macrostructure (example 05).

Example 05b: microstructure of scores of "SURFING GLOBAL CHANGE":

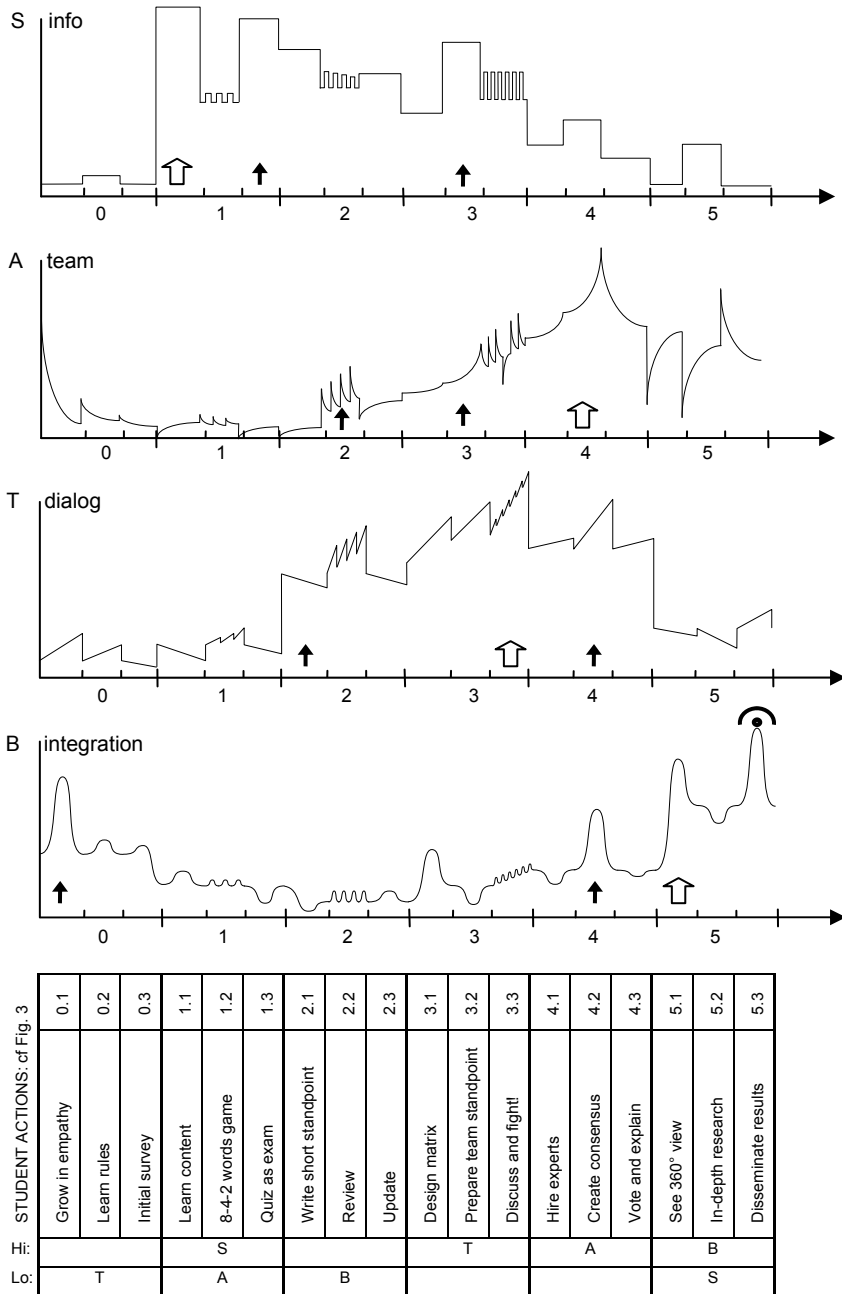


Fig. 14. Notation for a complex suite of game based learning: “SURFING GLOBAL CHANGE” - (“La partitura di SURFING GLOBAL CHANGE” © G. Ahamer) - Microstructure. Arrows: hi’s.

Each voice exhibits stages with high values (“hi”) and other stages with low values (“lo”), these are highlighted at the very bottom line of Fig. 14 and illustrated later in Fig. 15. SGC’s choreography in a certain sense represents a unit of a longer societal procedure: Level 5 (Bass) creates also new content for the subsequent year (= meaning of the music sign “fermate” at the end of the bass voice, meaning “hold this position”) which is the “final report” published on the public web outside the internal course web platform (several reports at IP, 2007) that serves as an additional input for the next year’s generation of IP students.

Table 3 explains the graphic elements (“notes”) for the four voices, dwelling on basic social and communicative characteristics of the respective dimensions.





voice	social meaning	note: graphic element	note description	symbolizes
Soprano	inform		containing block	inputted information, pushy
Alto	build team		relaxing impulse	is driven, remains when relaxing, hysteresis, structural memory, capacitor unloaded
Tenor	debate		increasing triangle	lancet, cutting, stiff, a shark’s fin, steady growth
Bass	integrate		sinusoid bell curve, Gaussian peak	equilibrating, sinusoidal peak, equilibrated effort, self-motivated and self-driven, coming and going, autopoietical ...

Table 3. Graphic elements referred to as “notes” for writing the 4 “voices” of the scores in SURFING GLOBAL CHANGE SGC.

When walking through examples 01 till 05, one could try to learn where the “optimum mixture” or “ideal equilibrium point” of a blend of different ingredients is, e.g. regarding social vs. academic learning. No! Replace this question for a “static equilibrium point” by asking for an optimal “rhythm of fluctuation” around such an idealized optimum, just as the bicycle driver oscillates around his intended path (chapter 3.3).

On a larger scale, one could even conceive a cyclic application of consecutive SGC-like processes, which makes the “hi’s” and “lo’s” of four voices appear in a repeated, even sinusoidal manner (see Fig. 15).

5.5 A first conceptual model of four dimensions in gaming and learning

As a next step of interpretation of the “four voices” pattern in a more stringent way than Table 2 and Fig. 8, Table 4 (above and below) puts them into two pairwise relationships in order to highlight that these four logically independent “social dimensions” are really suited to span up a complete vector space (as in vector calculus) for any human action.

voices	characteristic for both dimensions	its types of substrate are	logo constituted by	characteristic for both notes
S & T (☉ & ☺)	info space	facts	point(s)	angular = immediate disappearance at end
A & B (↔ & ↻ ↷)	person space	communication	arrow(s)	rounded = sinusoidal, gradually phasing out

voices	unit structure	its number is	logo type	note characteristic
S & A (☉ & ↔)	individual	one, the unit	onefold	steep inception
T & B (☺ & ↻ ↷)	compound (meta level in Table 5)	many, structure	manifold	gradual inception

Table 4. Pairwise similarities of substrates (above) and unit structures (below) of the four voices.

Table 5 deepens this systematization by attributing structural names such as “the one / many of facts / communications”, respectively. Similar to classical tarot cards (Waite, 2003), each “voice” or “dimension” represents a fundamental principle of life (like C.G. Jung’s archetypes, or planets in the interpretation of astrology) which can “assume a value” (i.e. be in one sign of the zodiac); just as in modern quantum mechanics, where (predisposed) values in state spaces are assumed by the so-called quantum numbers.

voices	logical structure (tarot card)	social meaning	graphical logo
S	“one of fact”	fact	☉
A	“one of communication”	team	↔
T	“structure of fact”	dialog	☺
B	“structure of communication”	integration	↻ ↷

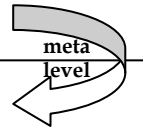


Table 5. Systematic description of the four voices : from “ones” to “structures”

When reiterating the entire above deliberation for the next (grand) cycle, i.e. the cycle of long-term societal procedures, the result “↻ ↷” will represent a new unit “☉” for the next iteration of structure building. Exactly here is the link from the “individual scale structurogenesis” (developed in this paper) to the “societal scale structurogenesis” (developed after decade-long experience with the author’s Global Change Data Base (GCDB, 2001).

Connection from social to global structurogenesis

This chapter 5 attempts to explore the question: how are structures created and generated? In a nutshell, SGC embarks on the following procedure of structurogenesis:

- First on the level of facts: (i) individually (ii) structured
- Second on the level of communication: (i) individually (ii) structured.

This evolutive pattern is exactly of the same nature as the GCDB trends suggest. Along global techno-socio-economic evolution and in an aggregated view of all economies of all single countries in the world, the following economic sectors are consecutively peaking (i.e. “carrying the melody and motive of evolution”) in their relative importance, starting from “agriculture” to “social services”:

- First on the level of matter and material: (i) individually (ii) structured
- Second on the level of infrastructure: (i) individually (ii) structured.

This temporal, but quite systemic order of growth phases could be hypothesized as the “STAB principle of global techno-socio-economic evolution”: In a structural sense, S resembles the sectors of agriculture and mining; T resembles commerce and manufacturing, A resembles construction and transport, B resembles civilizational infrastructure of electricity & gas and financial & social services (quantitative details in Ahamer, 2003: 10).

5.6 If you prefer mathematical language for models

The STAB temporal sequence $S \rightarrow T \rightarrow A \rightarrow B$ as described above is - in a mathematical sense - produced by a system of differential equations

$$S' \sim T; T' \sim A; A' \sim B; B' \sim -S$$

Where \sim equals “proportional” and $'$ stands for “first derivative with respect to time”.

A practical argument for the suitability of such a formal structure is the consecutive order of peaks in Fig. 15 which is idealized and generalized from Fig. 14 (lowest 2 lines with “hi’s” and “lo’s”).

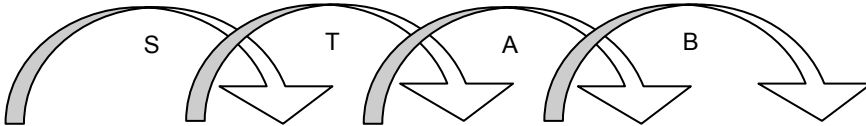


Fig. 15. The STAB principle in (individual or societal) learning

Mathematically, this structure of interrelations creates a four-dimensional standing wave in the four-dimensional state space of “learning”. Such structure is comparable with the two-dimensional standing electromagnetic wave following the mathematical structure $E' \sim H; H' \sim -E$, as in Maxwell’s classical electrodynamic equations (only involving a larger number of apparent variables). These two (electric E and magnetic H) fields produce a “standing wave” - the basis of all electronic communication.

In a symbolic sense, the “standing social wave” could be hypothesized as “optimal STAB learning path” and could relate to Csikszentmihalyi’s (1994) “flow” state.

5.7 Does the STAB procedure prevail in global techno-socio-economic evolution?

As already mentioned in chapter 5.5, the STAB sequence (Fig. 15) might also be the long-term structure for the evolving relative importance of economic sectors in all countries economics, as measured by the share of GDP ($\%GDP = \text{GDP}_{\text{sectoral}} / \text{GDP}_{\text{total}}$). Fig. 16 shows the average behavior of all countries’ economies as an idealized trend (each line is the share of one economic sector in percent: %GDP).

The definitions of S, A, T, B would additionally harmonize well with Manuel Castells' view of a *network society* pertaining to a "space of flows" which complements a "space of places":

- S = starting point in the *material world*: elements for "space of places"
- A = starting point in the *network society*: elements for "space of flows"
- T = evolutionary build-up in the *material world*: emerging structure in "space of places"
- B = evolutionary build-up in the *network society*: emerging structure in "space of flows"

In this light, civilisatoric evolution can be understood as a global learning process (Fig. 16).

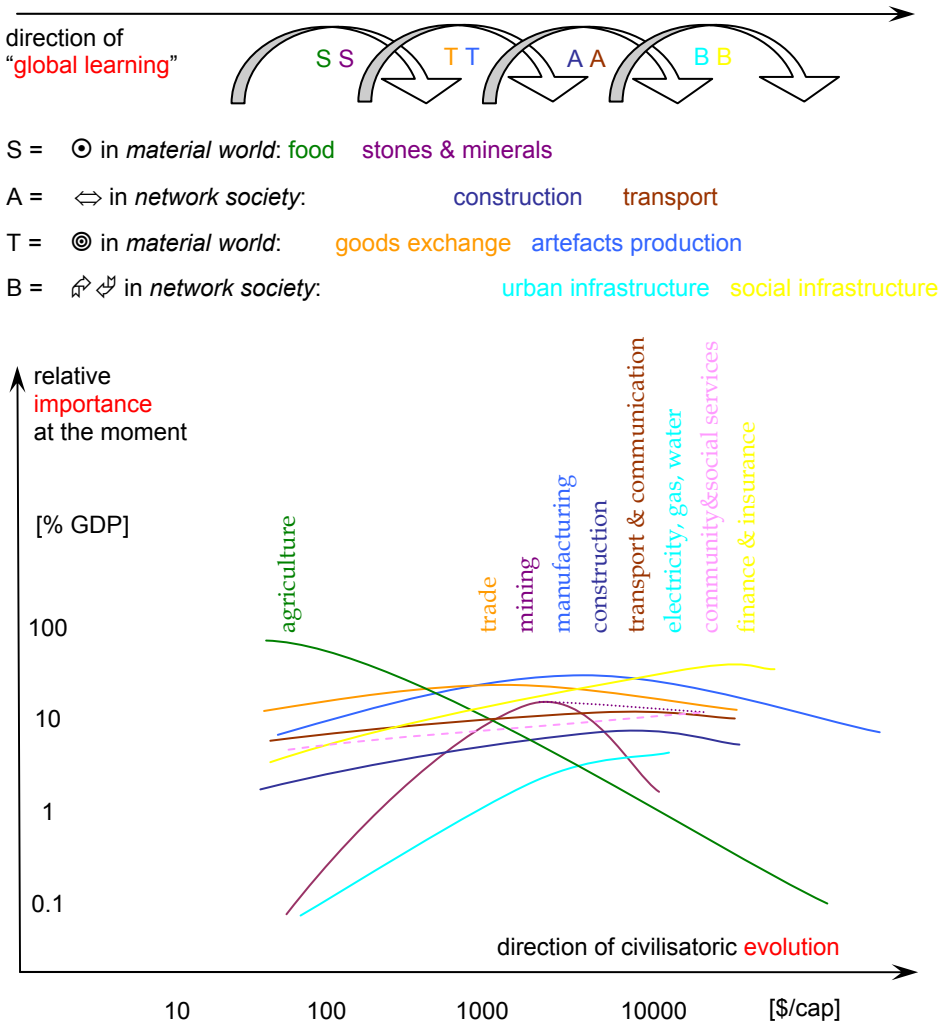


Fig. 16. The global techno-socio-economic evolution might follow the STAB principle. In this "map", each curve represents the relative importance of an economic sector as a function of economic level (GDP/cap.). Source: GCDB (2001), Ahamer (2003: 10, 2005: 103).

6. A quality criterion for effective learning procedures: STAB

SGC represents the “STAB strategy” of *informed team building and integration of opinions into a consensus*. This abbreviation recalls that the *most active voice* of the entire score is S, T, A, B, consecutively, along the time of growth. The notion of “progress” has been replaced by the notion of “growth” e.g. already by the American philosopher John Dewey (Berding, 2000; Pragmatism, 2007).

The idea of the “STAB strategy” is that *first* discussion should take place and *second* team structures will grow (not the other way round). (In principle, also the other sequence could occur, namely SATB – then team building occurs through personal and affective exchange – but given present experience, to the author this does not seem to be the main path.)

At any rate the procedure does *not run directly* from the individual sphere to integration! Apparently, humans need such intermediate and mediating steps (T, A) for creating a holistic view (B) on life affairs.

As a result of this paper, a quality criterion for gaming and learning can be expressed on two levels:

- (i) On the individual psychological level (micro-learning), a wisely designed and functioning STAB procedure could structurally resemble the “flow state” discovered by Csikszentmihalyi (1994) – which aphoristically could be called the “singing when working” criterion.
- (ii) As a practical consequence for the societal level (macro-learning), if institutions are conceptualized and structured in a non-dialogic manner (e.g. in a hierarchical manner) they could be seen as structurally suboptimally suited for responding to challenges – e.g. climate change. By inference, dialogic cultures (such as parts of the Anglo-Saxon culture with very old but still vivid democratic traditions, cf Landes, 1999; Moky, 1990) seem to be structurally better adapted to respond to severe structural global challenges, such as climate change.

For practical application of this general hypothesis, the art consists in arranging the social processes (sp) in a way that they create and give rise to an optimal societal procedure (SP) – which, in brief, is the task of a “social designer”.

Very generally and mathematically, *the optimal STAB design* would be defined by optimal growth of any of the four voices on the basis of a suitable rate of change and rhythm of the precursor voice.

7. Debriefing

It should be mentioned here that three levels of debriefing (Crookall, 1992; Lederman, 1992) were implemented for the case study “SURFING GLOBAL CHANGE” (SGC, 2006); in there a longer section on debriefing can be found:

1. inside SGC: regular feedbacks between students and facilitators during the game and at its end (upward semicircular arrows in Fig. 2)
2. on SGC: three written expert judgments from professionals on “game based learning” and “social dynamics” right after the first implementation in 2003
3. on SGC: official feedbacks from the side of the jury in the prize on media didactics “Medidaprix” (Medidaprix, 2007).

Additionally, inherent feedback activities occur in all levels based on the essence of SGC which is role switch between actor and monitoring spectator (cf. Petranek et al., 1992: 174). All three phases in the debriefing process (Lederman, 1992: 153) are thus operationalized. Summing up, the “debriefing as co-learner” (Stewart, 1992: 206), namely the facilitator, was the person learning most during the SGC implementations. He, the author, learned from those “who construct meaning”, i.e. the “learners” (Klabbers, 2000: 397).

8. Conclusion

Here “Design” is understood in a generalized way, as a “compound of temporal, spatial and interindividual rhythms enacting procedures” that may pertain to differing substrates (of which chapter 2 gave an overview).

It is useful to discern (i) individual micro-learning and (ii) societal macrolearning.

In order to map complex social, learning or societal procedures, the following four dimensions were introduced in this paper: information conveyed (“soprano”, S), team building (“alto”, A), dialogue of facts (“tenor”, T), and integration with others’ experience (“bass”, B). “Scores” can represent the choreographic structure of such procedures. One objective for writing such scores might be to safeguard their effectiveness for “real” learning.

After analyzing several years of experience with the negotiation game “SURFING GLOBAL CHANGE” (SGC), and more than a decade of work with practical procedures in climate protection measures and monitoring global evolution of techno-socio-economic structures (GCDB), the following *two generalized conclusions* are suggested (chapter 5.5):

- (1) If a communicational culture or an institutional landscape is constituted including a dialogic and team oriented component, chances are higher that such human cultures respond better to severe complex global challenges such as global warming.
- (2) If a societal procedure allows for the “STAB” sequence, it could be better apt for such complex evolutionary challenges. The “STAB” approach was developed in this paper and means that the *evolution of societal structures* follows this path: *increased learning of single facts generates and enhances dialogue which after growth generates team building that after growth prompts integration of differing views.*

The main question to be answered is: Which sequence of gaming framework conditions delivers an optimal learning effect (understood here as real change of behaviour), independently of the initial stage of mastery of the learners? – The answer is: *if structures are growing*. Namely, if a sufficient level of S (= information) produces a structure of T (= dialog), sufficient T produces a structure of A (= team), and sufficient A a B (= integration) structure. In other words, if the (rhythimized) learning sequence of S, T, A, and B finally produces adequate 360° worldviews for all participating individual learners, then the choreography and design were successful. Such findings are in principle supported by structural pattern analysis of economic global long-term trends, if societal evolution is understood as societal macro-learning process.

As a concluding hypothesis in a nutshell, this suggestion for “Managing Global Change” is delivered: National and supranational frameworks should allow for the “*STAB sequence of societal procedures*”, i.e. rhythmically include debate, team building and integration of opposing views. Resulting (societal) structure building facilitates sustainable consensus and hence is effective for a humane reality of living.

9. References

- ACCC (2007). Emission Trading. The Austrian Climate Portal, <http://www.accc.at>.
- Ahamer, G. (2003). Structural change in energy economics. Lecture notes for a Peak Oil Seminar (<http://math.uni-graz.at/keeling/peakoil/dokumentation.html>), Graz Univ., http://www.uni-graz.at/globalstudies/deposit/USW_VisionWirtschaftsEnergiekapitel.pdf.
- Ahamer, G. (2004). Negotiate your Future: Web Based Role Play. *Campus-Wide Information Systems*, 21(1), 35-58.
- Ahamer, G. (2005). Imposing a Dialogue Helps to Minimize a “Clash of Cultures”. In Tschandl, M. (Ed.), *The Challenge of the EU Enlargement* (pp. 35-63). Graz: Leykam.
- Ahamer, G. & Rauch, H. (2006). From “Vertical” to “Horizontal” Didactics – Generating a Tissue of Perspectives of “Global Change”. *Zeitschrift für Hochschulentwicklung* 1(2), 127-147, <http://www.zfhe.at>.
- Ahamer, G. & Schrei, C. (2006). Exercise ‘Technology Assessment’ through a gaming procedure. *Journal of Design Research*, 5(2), 224-252.
- Akademie (2006). 3x7=21: A discursive structure for quality enhancement in interdisciplinary university courses. Academy for New Media and Technology Transfer http://www.uni-graz.at/anmwww/anmwww_fachbereiche/anmwww_mediendidaktik.htm, or <http://www-gewi.uni-graz.at/cocoon/mdm/text?pid=anmw-mdm-3mal7ist21>, Graz University.
- Aschemann, R. (2004). Lessons learned from Austrian SEAs. *European Environment*, 14(3), 165-174.
- Badke-Schaub, P. (2004). Strategies of experts in engineering design: between innovation and routine behaviour. *Journal of Design Research*, 4(2).
- Bechmann, G., Decker, M., Fiedeler, U., Krings, B.J. (2007). TA in a complex world. *International Journal of Foresight and Innovation Policy*, 3(1), 6-27.
- Beilharz, K. (2004). Designing Sounds and Spaces – Interdisciplinary Rules & Proportions in Generative Stochastic Music and Architecture. *Journal of Design Research*, 4(2).
- Berding, J.W.A. (2000). John Dewey's participatory philosophy of education – Education, experience and curriculum. The history of education and childhood, Nijmegen University, <http://members.ziggo.nl/jwa.berding/Summarydiss.doc>.
- Bianconi, F., Saetta, S.A., Tiacci, L. (2006). A web-based simulation game as a learning tool for the design process of complex systems. *Journal of Design Research*, 5(2), 253-272.
- Bilda Z., Gero, J.S., Purcell, T. (2006). To sketch or not to sketch – That is the question. *Design Studies*, 27(5), 587-613.
- Bossel, H. (2004). *Model Building and Simulation*. Vieweg, Braunschweig.
- Bouchard, C., Aoussat, A., Duchamp, R. (2006). Role of sketching in conceptual design of car styling. *Journal of Design Research*, 5(1), 116-148.

- Boujut, J.-F., Tiger, H. (2002). A Socio-Technical Research Method for Analyzing and Instrumenting the Design Activity. *Journal of Design Research*, 2(2).
- Casakin, H. (2004). Visual Analogy as a Cognitive Strategy in the Design Process. Expert Versus Novice Performance. *Journal of Design Research*, 4(2).
- Corbeil, P. (2005). History and the Computer in Canadian Institutions: An Overview. *Social Science Computer Review*, 23(5), 181-189.
- Corbusier (2007). Fondation Le Corbusier. <http://www.fondationlecorbusier.asso.fr>.
- Couvent de la Tourette (2006). <http://www.couventlatourette.com>, last retrieved 3.2.06.
- Crookall, D., Bradford, W. (2000). Impact of climate change on water resources planning. *Proceedings of the Institution of Civil Engineers - Civil Engineering*, 138(2), 44-48.
- Crookall, D. (2003). The Art and Science of Design. *Simulation & Gaming - An International Journal*, 34(4), 485.
- Crookall, D. (2003a). International Relations and Simulation. *Simulation & Gaming - An International Journal*, 34(4), 221-225.
- Czikszentmihalyi, M. (1994). *Flow: the psychology of optimal experience*. Nightingale-Conant, New York.
- Daly, H. (1999). *Beyond Growth: The Economics of Sustainable Development*. Beacon Press.
- Decker, M. (2001). *Implementation and Limits of Interdisciplinarity in European Technology Assessment*. Berlin, Springer.
- Decker, M., Ladikas, M. (2004). *Bridges between Science, Society and Policy. Technology Assessment - Methods and Impacts*. Berlin, Springer.
- Dong, A. (2007). The enactment of design through language. *Design Studies*, 28(1), 5-21.
- Dorst, K., Cross, N. (2001). Creativity in the design process: co-evolution of problem-solution. *Design Studies*, 22(5), 425-437.
- EC (2007). Pre-Accession Assistance for Institution Building - Twinning. See http://ec.europa.eu/enlargement/financial_assistance/institution_building/twinning_en.htm and Twinning Brochure "Building Europe Together", see http://ec.europa.eu/enlargement/pdf/twinning_brochure_2005_en.pdf
- ESD (2007). Motivation and Guiding Idea of the NGO "European Association for the Promotion of Sustainable Development" ESD, Vienna, see <http://www.esd-eu.org>.
- GCDB (2001). A Structured Basket of Models for Global Change. In C. Rautenstrauch and S. Patig (Eds.) *Environmental Information Systems in Industry and Public Administration* (pp.101-136). Hershey: Idea Group Publishing.
- Grunwald, A. (1999). Technology Assessment or Ethics of Technology? Reflections on Technology Development between Social Sciences and Philosophy. *Ethical Perspectives*, 6(2), 170-182.
- GS (2007). Global Studies: Master Curriculum. <http://www.uni-graz.at/globalstudies>.
- Heaton, L. (2002). Designing Work. Situating Design Objects in Cultural Context. *Journal of Design Research*, 2(2).
- HdZ (2007). Research Programme "House of the future" by the Austrian Federal Ministry for Science and research, <http://www.hausderzukunft.at>.
- Hellinger, B. (2000). *Ordnungen der Liebe*. Knauer. (In English: The Art and Practice of Family Constellations Leading Family Constellations as Developed by Bert Hellinger, ed Ulsamer, B., Tucker & Theisen, 2003).
- Herder, P.M., Turk, A.L., Subrahmanian, E., Westerberg, A.W. (2003). Collaborative Learning in a Cross-Atlantic Design Course. *Journal of Design Research*, 3(2).

- Hofmeyer, H., Rutten, H.S., Fijneman, H.J. (2006). Interaction of spatial and structural design, an automated approach. *Design Studies*, 27(4), 423-438.
- Hofstede, G. (1994). *Cultures and Organisations*. HarperCollinsBusiness, London.
- Huizinga, J. (1994). *Homo Ludens*. Beacon Press.
- IP (2005). World in a change? Comparison of three socio-economic, climatic and technologic perspectives of the future. *Interdisciplinary Practical*, see http://www.uni-graz.at/en/usw1www_le_studreinfo_ip_weltwandelkurzinfo.pdf.
- IP (2007) = Passive Houses and Low Energy Houses. Final Report, http://www.uni-graz.at/usw1www/usw1www_publicationen/usw1www_berichte.htm, USW Report 2006/01.
- IPCC (2007). Fourth Assessment Report. www.ipcc.ch and <http://ipcc-wg1.ucar.edu/>
- Jarvenpaa, S.L. and Leidner, D.E. (1998). Communication and Trust in Global Virtual Teams, *Journal of Computer-Mediated Communication*, 3(4), see <http://jcmc.indiana.edu/>
- JDR (2006). Special Issue on the use of games in and for design. *Journal for Design Research* 5(2), 149-154, see <http://www.inderscience.com>.
- Johns, R., Shaw, J. (2006). Real-time immersive design collaboration: conceptualising, prototyping and experiencing design ideas. *Journal of Design Research*, 5(2), 172-187.
- KEK (1997). Municipal Energy Concept of the City of Graz, Report Series "KEK-Reports", <http://www.oekostadt.graz.at/cms/beitrag/10084666/1624842/>.
- Klabbers, J.H.G. (2000). Learning as acquisition and learning as interaction. *Simulation & Gaming: An Interdisciplinary Journal*, 31(3), 380-406
- Klabbers, J.H.G. (2003). Gaming and simulation: Principles of a science of design. *Simulation & Gaming: An Interdisciplinary Journal*, 34(4), 569-591.
- Kratena, K., Schleicher, S., Friedl, B., Schnitzer, H., Gartner, H., Jank, W., Ahamer, G., Radunsky, K. (1998). The Toronto Technology Program. Austrian Council on Climate Change, November 1998, <http://www.accc.at/pdf/ttp98.pdf>.
- Kristjánsson, K. (2006). Emotional Intelligence in the Classroom - An Aristotelian Critique. *Educational Theory*, 56(1), 39-56.
- Landes, D.S. (1999). *The Wealth and Poverty of Nations. Why Some Are So Rich and Some So Poor*. Norton, New York.
- Lloyd, J.R. (2004). INTEnD: A Dispersed, Virtual Engineering Design Team Approach to Globalize Engineering Education. *Journal of Design Research*, 4(4).
- Lourdel, N., Harpet, C., Laforest, V., Gondran, N., Brodhag, C. (2006). Sustainable development training by simulation of an industrial crisis. *Journal of Design Research*, 5(2), 188-200.
- LRP (1995). Graz Municipal Plan for Air Hygiene, <http://www.feinstaubfrei.at/down/LRP-ZF.pdf>
- MacGregor, S. (2002). New Perspectives for Distributed Design Support. *Journal of Design Research*, 2(1).
- Maher, M.L., Tang, H-H. (2003). Co-evolution as a computational and cognitive model of design. *Research in Engineering Design*, 14(1), 47-63.
- Mayer, I., Veenemann, W. (2002). Games in a World of Infrastructures. *Simulation-games for Research, Learning and Intervention*. Eburon, Delft.

- Mayer, I.S., van Daalen, C.E., Bots, P.W.G. (2004). Perspectives on policy analyses: a framework for understanding and design. *Int. J. Technology, Policy and Management*, 4(2), 169-191.
- Mayer, I.S., Bockstael-Blok, W., Valentin, E.C. (2004). A Building Block Approach to Simulation: An Evaluation Using Containers Adrift. *Simulation & Gaming - An International Journal*, 35(3), 29-52.
- Mayer, I.S., van Bueren, E.M., Bots, P.W.G., van der Voort, H. (2005). Collaborative decisionmaking for sustainable urban renewal projects: a simulation-gaming approach. *Environment and Planning B: Planning and Design*, 32, 403-423.
- Medidaprix (2007). Finalists of the prize on media didactics.
http://www.medidaprix.org/mdd_2007/dynframeset_006.html.
- Mokyr, J. (1990). *The Lever of Riches: Technological Creativity and Economic Progress*. New York: Oxford University Press.
- Moser, I. & Moser, F. (2005). *The Disappearance of the Universe*. Graz.
- Ono, M.M. (2006). Cultural diversity as a strategic source for designing pleasurable and competitive products, within the globalisation context. *Journal of Design Research*, 5(1), 3-15.
- Ossimitz, G. (2000). *Development of Systemic Thinking. Theoretical Concepts and empirical Investigations*. Klagenfurt University, <http://www.uni-klu.ac.at/~gossimit>.
- Pilch, B., Aschemann, R., Ahamer, G. (1992). Luftreinhaltung in der Stadtökologie - eine Systemanalyse. *Mitteilungen des Naturwissenschaftlichen Vereins Steiermark*, 122, 19-27.
- Popov, L.S. (2002). Architecture as Social Design. The Social Nature of Design Objects and the Implications for the Profession. *Journal of Design Research*, 2(2).
- Pragmatism (2007). Dewey as founder. <http://en.wikipedia.org/wiki/Pragmatism>.
- Prensky, M. (2001). *Digital Game-Based Learning*. McGraw-Hill, New York, 2001.
- Restrepo, J., Christiaans, H. (2004). Problem Structuring and Information Access in Design. *Journal of Design Research*, 4(2).
- Raskin, P. et al. (2002). *The Great Transition*. Stockholm Environment Institute, <http://www.sei.se>.
- Roth, W.-M., Lawless, D.V., Masciotra, D. (2001). Spielraum and Teaching. *Curriculum Inquiry*, 31(2), 183-207.
- Schrei, C. (2006). Minimal. Master Thesis, see: <http://www.nook.at/minimal/>.
- Schrei, C. (2007). Information design and graphic design. See: <http://www.definite.at/>.
- SGC (2006). Surfing Global Change: Negotiating sustainable solutions. *Simulation & Gaming - an International Journal*, 37(3), p. 380-397. <http://sag.sagepub.com>.
- SI (2007). Austrian-Dutch submission for the Twinning SI2006/IB/EN/01 "Development of financial instruments for water management based on Water Framework Directive 2000/60/EC", Federal Environment Agency Vienna.
- SiP (2004). Didactics: Are we in the trend? - Is the trend in us? *Journal of the Society of Living with Children*, 36 (May 2004), 6-9, directly at http://knallerbse.world4you.at/files/schulzeitung/schulzeitung_36_artikel_2.pdf or see Schule - Schulzeitung - Archiv at <http://www.knallerbse.at>.
- Temple, J. (1999). The New Growth Evidence. *Journal of Economic Literature*, 37(1), 112-156.
- UBA (2004). Greenhouse gases and climate change, http://www.umweltbundesamt.at/fileadmin/site/umweltkontrolle/2004/E0601_klima.pdf

- UBA (2007). UVP Dokumentation. Genehmigung & Feststellung, see <http://www.umweltbundesamt.at/umweltschutz/uvpsupemas/uvpoesterreich1/uvpdatenbank>.
- UNFCCC (2007). United Nations Framework Convention on Climate Change, see <http://unfccc.int>.
- USW (2007). Interdisciplinary Practicals in the curriculum of Environmental Systems Science http://www.uni-graz.at/en/usw1www/usw1www_lehre/usw1www_lehre_ip.htm, (USW = Umweltsystemwissenschaften).
- Simulation & Gaming (2007). Ready-to-use simulation games. Section in the highly ranking international journal S&G, see <http://www.unice.fr/sg> or <http://sag.sagepub.com>.
- SGC (2006). SURFING GLOBAL CHANGE: Negotiating sustainable solutions. *Simulation & Gaming - an International Journal*, 37(3), 380-397, see <http://sag.sagepub.com>.
- SonEnvir (2007). Sonification Environment: a sonification project. <http://sonenvir.at>.
- Van Bueren, E., Van Der Voort, H. Maas, N. (2006) The Sureuro Gaming Exercise: designing a game for sustainable refurbishment by housing companies. *Journal of Design Research*, 5(2), 201-223.
- Vester, F. (1980). Sensitivitätsmodell. Frankfurt/Main.
- Vocal Consort (2007). Score Isaac, <http://www.vocalconsort.at/repertoireframeset.htm>
- Waite, A.E. (2003). Universal Waite Tarot Deck with Book. US games.
- WegCenter (2007). An Inconvenient Truth: Austria and the Kyoto Target, http://www.uni-graz.at/en/igam7www/igam7www_aktuell/igam7www_oeffentliche_statements.htm.
- Weick, K.E. (1979). *The social psychology of organizing*. Reading: Addison-Wesley.
- Weinberg, A. (2007). Psychodrama & sociodrama. <http://www.alfred-weinberg-psychodrama.de> or <http://www.psychodrama-deutschland.de>.
- Wodehouse, A., Bradley, D. (2006). Gaming techniques and the product development process: commonalities and cross-applications. *Journal of Design Research*, 5(2), 155-171.
- Xenakis, I. (2007). Selection of compositions. <http://www.iannis-xenakis.org>.
- Zangemeister, D. (1970). *Nutzwertanalyse in der Systemtechnik - Eine Methodik zur multidimensionalen Bewertung und Auswahl von Projektalternativen*, Wittmann.
- Zermeg III (2007). Decision guidance for the visualisation of the achievements of sustainable corporate strategies. Final report to the Austrian Ministry of Science, <http://www.fabrikderzukunft.at>.
- Zographos, M. (2007) Iannis Xenakis: the aesthetics of his early works. <http://www.furious.com/perfect/xenakis.html>.

Heuristics and pattern recognition in complex geo-referenced systems

Gilbert Ahamer, Adrijana Car, Robert Marschallinger,
Gudrun Wallentin and Fritz Zobl

*Institute for Geographic Information Science at the Austrian Academy of Sciences
ÖAW/GIScience, Schillerstraße 30, A-5020 Salzburg, Austria*

Abstract

We propose different methods to represent “time” in geographic and other perceptions of reality. One appropriate method is to show effects of the procedures which take “place” along the continuum of “time”. Similarly, perspectives on reality can be “mapped” – such is the core of geography.

This presents a special opportunity for IT to develop tools for various disciplines which are then interchangeable. IT allows views on new worlds. These worlds arise by applying new perspectives to known reality. IT helps to organise the complexity of the resulting views. IT creates images of reality. IT is able to move from “Geographic Information Science” to “Interperspective Information Science”.

Additionally, it is able to host negotiation processes that generate new spaces of understanding created by consensus building.

The above two tasks for IT creatively contribute to building a “network society”.

Key Words: Geographic Information Science (GIS), mapping, time, space, perception, consensus building, social space.

1. Let's start to think

1.1 Our world is the entirety of perceptions. (Our world is not the entirety of facts.)



Fig. 1. At left: The human being perceives the world. At right: The “primordial soup” of living, before the advent of (social) organisms: uncoordinated perspectives, uncoordinated world views.

Hence, every individual lives in a different world (Fig. 1 at left).

1.2 The “indivisible unit”, the atom ($\alpha\tau\omicron\mu\omicron\varsigma^1$) of reality, is equal to one (human) perspective. Our world is made up of a multitude of perceptions, not of a multitude of realities and not of a multitude of atoms (Fig. 1 at right).

1.3 In order to share one’s own conception with others, “writing” was invented. Similarly, complex structures, such as landscapes, are “mapped”. To map means to write structures.

1.4 Writing helps to become aware. We ask: Is it possible to map = write

1. the distribution of material facts and elements in geometric space? (physics)
2. the distribution of factual events in global time? (history)
3. the distribution of real-world objects across the Earth? (geography)
4. the distribution of elements along material properties? (chemistry)
5. the distribution of growth within surrounding living conditions²? (biology)
6. the distribution of persons acting in relationships? (sociology)
7. the distribution of individuals between advantage and disadvantage of trade? (economics)
8. the distribution of perspectives within feasible mindsets? (psychology)
9. the distribution of living constructs along selectable senses? (theology)

We see: awareness results from reflection (Fig. 2).

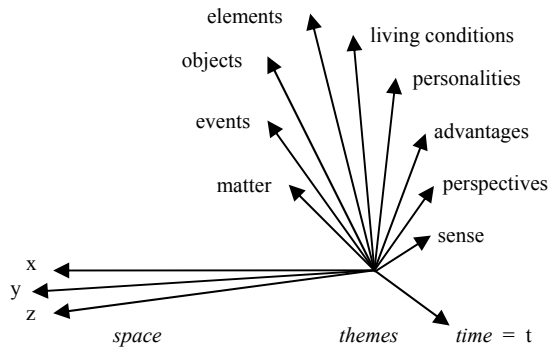


Fig. 2. Fundamental dimensions, along which to coordinate individual world views when reflecting.

2. Time can be

1. an attribute of space (a very simple historic GISystem)
2. an independent entity (Einstein’s physics)
3. the source of space (cosmology).

In terms of GIS item 2.1 is expressed as “t is one of the components of geo-data”³ (Fig. 3).

¹ what cannot be split any further (Greek)

² životné prostredie (Slovak): living environment

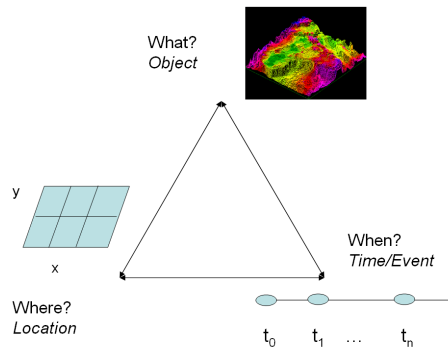


Fig. 3. The where-what-when components of geo-data, also known as triad (Peuquet 2002: 203).

Time can be understood as

- establishing an ordinal scale for events
- driving changes (= Δ) of realities
- something that unfortunately does not appear on paper.

A proposed solution is to map changing realities (Δ) instead of mapping time. Time is replaced by what it produces. This is indicated in Fig. 4.

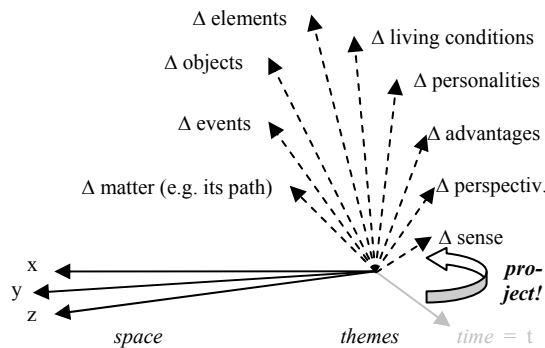


Fig. 4. The projection of time (t) onto the effects of time (the changes Δ) can apply to any science.

This idea flips = projects the t axis onto one of the vertical axes. Time means then: how maps are changed by the envisaged procedures. Such procedures modify the variables along the axes, be they of physical (gravity force) or of social nature (war).

A classical example is Minard’s map of Napoleon’s 1812 campaign into Russia⁴ (Fig. 5a, b).

³ GIScience goes way beyond this view of time and space (considering time as function) because it allows for much more complex queries and analyses.

⁴ Patriotic War (in Russian): Отечественная война

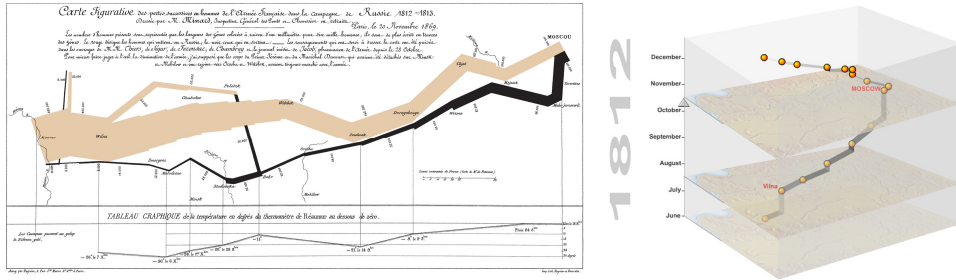


Fig. 5. Notions of path in a geo-space: (a) Minard’s map of human losses during Napoleon’s 1812 campaign into Russia; and (b) its geovisualisation in a time cube (Kraak, 2009).

Further examples such as landslides in geology, growth of plants, energy economics, economics will be shown in chapter 7.

For implementing the idea to project the t axis onto the Δ axis we need to have clear insight how time quantitatively changes reality.

In other words: we need a model, which (explaining how processes occur) determines the representation of time (Fig. 6). Examples are sliding geology, Δ GDP/cap, plant growth.

One cannot perceive time (never!), only its effects: what was perceived in this time span (duration)⁵? This is why the t axis is projected onto another axis denoting the effect of elapsed time; what this means to the individual sciences is shown in Fig. 4.

Very similarly, in physics nobody can feel force, only its effect (deformation, acceleration), and still forces have been undisputedly a key concept for centuries.

*What is time? Just a substrate for procedures.
What is space? Just hooks into perceived reality.*

We retain from this chapter 2 that we need a clear model of how elapsing time changes reality. Then we can map time as suggested: by its effects.

3. How to write time?

The big picture shows us various examples:

1. as a wheel (see the Indian flag): revolving zodiacs, rounds in stadiums, economic cycles, Kondratieff’s waves
2. as an arrow (see Cartesian coordinates): directed processes, causal determinism, $d/dt, d^2/dt^2$
3. as the engine for further improvement (evolutionary economics): decrease vs. increase in global income gaps, autopoietic systems, self-organisation
4. as the generator of new structures (institution building, political integration, progressive didactics): new global collaborative institutions, peer-review, culture of understanding, self-responsible learning, interculturality
5. as evolving construct (music).

⁵ T. de Chardin’s (1950) concept of *durée* (French).

From this chapter 3 we only keep in mind that the concepts to understand and represent time are fundamentally and culturally different.

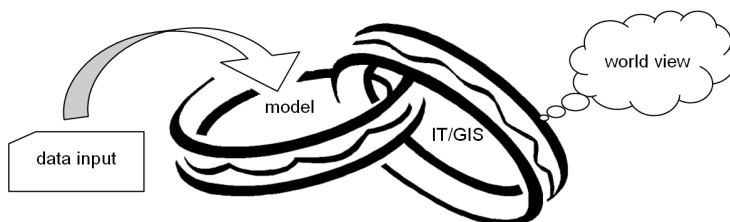


Fig. 6. All data⁶ representations require models.

4. How to write space?

The big picture shows us various examples:

1. as a container of any fact and any process (geography and GIS)
2. as result of human action (landscape planning)
3. as evolving construct (architecture).

Examples span space as

- received and prefabricated versus
- final product of one's actions, namely:
 1. spaces as the key notion for one's own science: everything that can be geo-referenced means GIS
 2. space as the product of human activity
 3. expanding space into state space: the entirety of possible situations is represented by the space of all "state vectors" which is suitable only if procedures are smooth.

The main thesis here is: the "effects of time" are structurally similar in many scientific disciplines, and they often imply "changes in structures" too. Information Technology (IT) is already providing scientific tools to visualise such structures.

5. How to map space and time?

The detailed picture: it is obvious that a choice must be made for one mode of representation and for one view of one scientific discipline:

1. (x, y, t) : cartography, GIS (Fig. 7)
2. (x, y, z, t) : geology
3. $(x, y, z; v_x, v_y, v_z, t)$: landslides
4. $(x, y, z; \text{biospheric attributes}; t)$: ecology, tree-line modelling
5. (countries; economic attributes; GDP/cap) or (social attributes; structural shifts; elapsing evolutionary time): economic and social facts in the "Global Change Data Base"⁷ (Fig. 8)
6. perceiving rhythms and structures: (only) these are "worth recognising": music, architecture, fine arts.

⁶ datum (Latin): what is given (unquestionable)

⁷ This GCDB is described in Ahamer (2001)

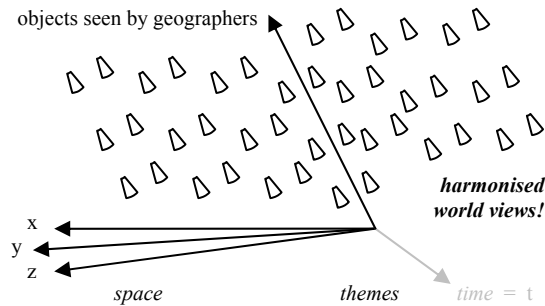


Fig. 7. Harmonising world views: GIS reunites world views by relating everything to its location.

Different sciences may have considerably different outlooks on reality (Fig. 8). A humble attitude of recognising facts⁶ instead of believing in the theories one's own discipline offers can empower people to survive even in the midst of other scientific specialties: Galileo's (1632) spirit: give priority to observation, not to theories!

This is the essential advantage of geography as a science: geographers describe realities, just as they appear. Such a model-free concept of science has promoted the usefulness of GIS tools to people independent of personal convictions, scientific models or theories.

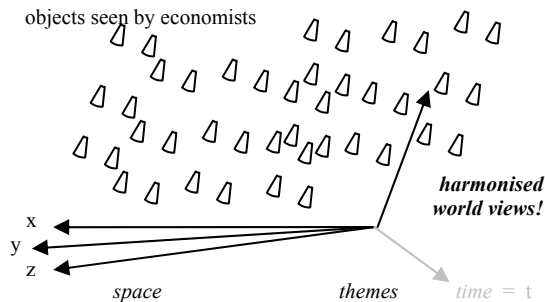


Fig. 8. Different but again internally harmonised world views: explain facts from another angle.

6. What IT does, did, and could do

6.1 IT helps to organise the multitude of views (= perceptions) onto data that are generated by humans:

- IT constructs world views, such as: GIS, history, economics, geology, ecology etc.
- IT has already largely contributed to demolishing traditional limitations of space and time:
 - Space: tele(-phone, -fax, -vision), virtual globes (Longley et al., 2001)
 - Time: e-learning, asynchronous web-based communication, online film storage (Andrienko & Andrienko 2006).

6.2 This paper investigates non-classical modes of geo-representation.

We would like to point out that there are two already well-established fields that offer solutions to mapping (space and time, Fig. 9) views: Scientific and information visualisation are branches of computer graphics and user interface design which focus on presenting data to users, by means of interactive or animated digital images. The goal of this field⁸ is usually to improve the understanding of the data presented. If the data presented refers to human and physical environments, at geographic scales of measurement, then we talk about Geovisualisation, e.g. (MacEachren, Gahegan et al. 2004; Dykes, MacEachren et al. 2005, Dodge et al., 2008).

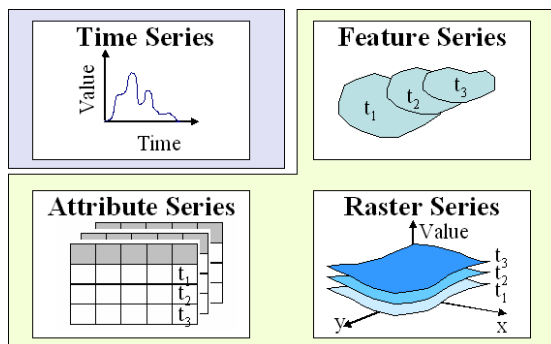


Fig. 9. Time series and three spatio-temporal data types (<http://www.crrw.utexas.edu/gis/gishydro05/>).

6.3 IT could develop tools that are then interchangeable across scientific disciplines, e.g. landslides that may structurally resemble institutional and economic shifts (see 7.1). IT could prompt scientists to also look at data structures from other disciplines. Whatever the disciplines may be, the issues are structures and structural change!

7. Examples

The authors are members of the “Time and Space” project at their institution named “Geographic Information Science”⁹, a part of which explores the cognitive, social, and operational aspects of space & time in GIScience.

This includes models of both social and physical space and consequences thereof for e.g. spatial analysis and spatial data infrastructures. We investigate how space and time are considered in these application areas, and how well the existing models of space and time meet their specified needs (see e.g. Fig. 9 left). This investigation is expected to identify gaps. Analysis of these gaps will result in improved or new spatio-temporal concepts particularly in support of the above mentioned application areas.

⁸ http://en.wikipedia.org/wiki/Scientific_Visualization

⁹ The overarching aim of the GIScience Research Unit is to integrate the “G” into Information Sciences (GIScience, 2009)

7.1 Sliding realities: geology

The notion of the path in geography (x, y, t) is extended by the z axis (see item 5.2) which produces a map of “time”: Fig. 9 left (Zobl, 2009).

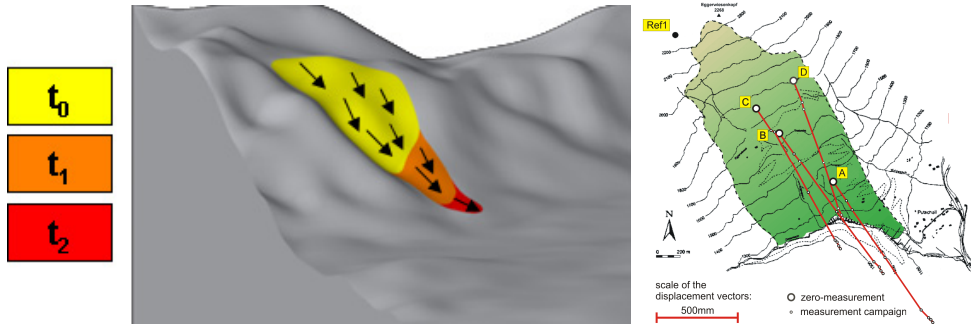


Fig. 9. Left: Geology takes the $(x, y, z; t)$ world view. Right: These effects of time occur in space, most helpfully. Source: Brunner et al. (2003).

The “effect of time” is sliding (luckily in the same spatial dimensions x, y, z): we take the red axis in Fig. 9 right. Space itself is sufficiently characteristic for denoting the effects of time.

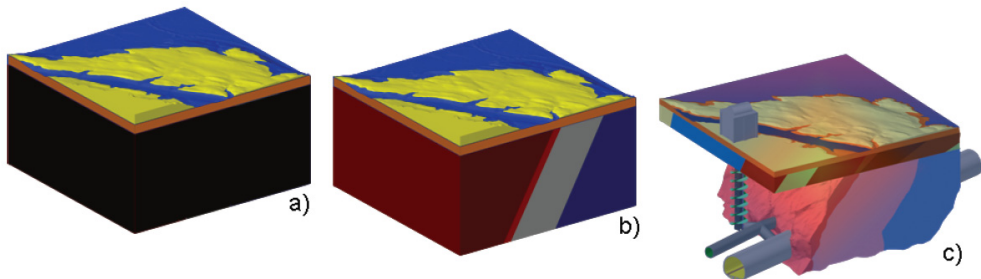


Fig. 10. This series of understandings of bedrock shows how data are stepwise combined with model results in order to reach approximation of understandings (Klima & Zobl, 2009).

7.2 Slices of realities: geology

Despite the lucky coincidence that the effect of time ($\Delta x, \Delta y, \Delta z$) occurs in the same space (x, y, z) we try to produce slides carrying more information (item 5.3) and hence recur to the so-called attributes mentioned in Fig. 9 such as grey shades or colours.

The speed of sliding ($d/dt x, d/dt y, d/dt z$) is denoted both by horizontal offsets and whitish colours in the spaghettis (Marschallinger, 2009) of Fig. 11.

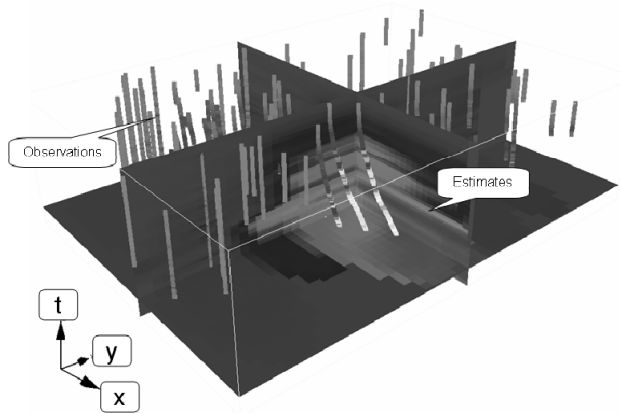


Fig. 11. The $(x, y, z; v_x, v_y, v_z; t)$ view of a landslide process (shades of grey mean speed v).

7.3 Slide shows

How to map spatial realities that are not any longer isotropic displacement vectors of space itself? For the example of changing tree lines in the Alps (Wallentin, 2009) a slide show is used to present the change of growth patterns made up of the multitude of individual agents (= trees = dots in Fig. 12). Moving spatial structures are depicted as a film of structures (item 5.4).

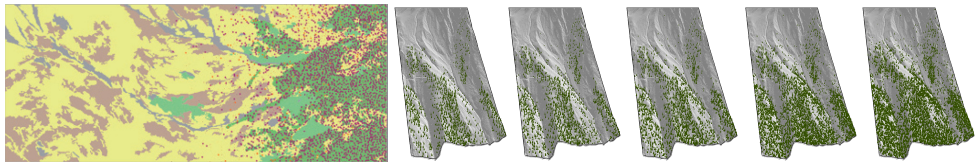


Fig. 12. The $(x, y, z; \text{biospheric attributes}; t)$ view of the Alpine tree line (above) and its shift induced by climate change as a slide show (below), as computed by the TREELIM model.

In such processes which involve independent behaviour of autonomous agents (here: trees) it becomes seemingly difficult to apply a transformation of space itself, e.g. $d/dt(x, y, z)$.

The model used and its background is briefly described: alpine tree line responses dynamically to changes in the environment. Currently a substantial upwards shift of the alpine tree line can be observed due to land use and climate change. The spatio-temporal tree patterns are modelled in an individual-based approach, where system-level attributes emerge from ecological processes of single trees, their mutual interactions and reactions to environmental factors such as climate or the elevation gradient.

TREELIM is an individual based model that was developed to get a better understanding of the alpine tree line dynamics in respect to land-use change. The model was validated for a case study in Längenfeld, Ötztal (Tyrol, Austria) over a period of 52 years (Wallentin et al. 2008). Individual based models that are structurally realistic model the real-world processes

that drive landscape patterns (Grimm & Railsback 2005). Thus TREELIM is designed as a generic model that can be extrapolated in space and time.

Traditionally, individual based models are validated through a model-reality comparison of spatial patterns at a certain point in time. However, a dynamic system does not merely have characteristic spatial patterns, but rather can be described through spatio-temporal patterns. Whereas in non-spatial process models of ecosystems, the temporal aspect is commonly considered as a crucial point in the model validation, this is not the case for spatial models. For spatial models, i.e. models that result in maps as the central model output, the model validation focuses on spatial aspects at a certain point in time (a snapshot situation). Although the temporal aspect may be included to some extent by considering time steps, (i.e. distribution of a spatial feature at several points in time) there is no consideration of spatio-temporal patterns and temporal model uncertainties in the model output variables, as e.g. the average tree line elevation.

7.4 Global deforestation

One key driver for global change is deforestation; easy to map as change of land use category of a given area (Fig. 13).

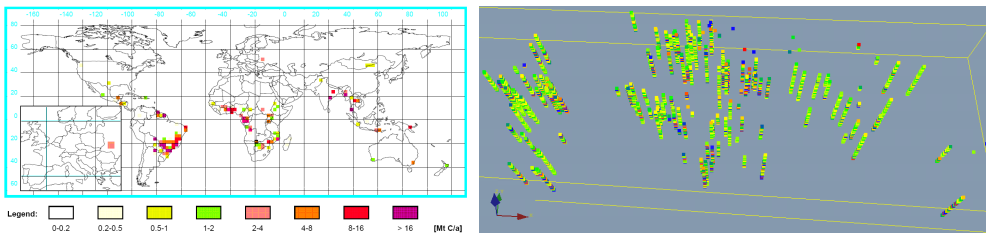


Fig. 13. The $(x, y, z; \Delta$ biospheric attributes; t): view of the global deforestation process in megatons carbon. Above: map of carbon flow, below: time series of GCDB data per nation symbolically geo-referenced by the location of their capitals.

This representation is analogous to Fig. 11. In both, the focus shifts from maps(t) \rightarrow maps($t, \Delta t$). Interest includes temporal dynamics:

t = colour (above); Δt = height+colour (below), enriching the purely spatial interest.

Even if to the aim is to enlarge the scope of the information delivered from the static map (Fig. 13 above) to the “dynamic map” (Fig. 13 below), readers will remain unsatisfied because no insight into the dynamic properties of deforestation is provided (Fig. 18).

Increasingly, the viewer’s focus turns further from “facts” to “changes of facts”, to “relationships with driving parameters¹⁰” and to (complex social and political) “patterns¹¹”.

7.5 Space as text

Studying geo-referenced data sets (GIS) can help to facilitate bridging interperceptual gaps. Critical GIS literature (e.g. Sarah Elwood) suggests that GIS tools are combined with

¹⁰ see the suggested scenarios for water demand, water supply and water quality (Ahamer, 2008)

¹¹ Patterns: name of the journal of the American Society for Cybernetics ASC

gaining power over space (Fischer, 2008a). Theories going back to Ferdinand de Saussure (Fischer, 2008b) understand space as “text”. Societies write space and then read it. GIS systems are the expression of the ability to “write space”, they express the “space views of those being able to write space” (Fischer, 2008c).

7.6 Spaces constituted by social media

The concept of Manuel Castell’s “space of flows” (Castells, 2004, Ahamer & Strobl, 2009) sees space as constitutes by communication. Urban planning (Crang, 2000) sees that “electronic media has raised issues of political action, community formation and changing identities”. “The metaphorical adoption of urban models is co-determined by electronic sociality and suggests four principle approaches: cities set in or against world flows, suburbanised telecities, communitarian visions and accounts that appeal to a renewed public sphere – all of them shape an electronic architecture. Spatial metaphors and electronic practices are seen as entangled and shaping each other.” “In this sense, the ‘real’ city is the indefinable complexity and folding of spaces—lying outside the visualisations offered of cyberspace.”

Urban social relations are co-determined by (electronic) communication (Purcell, 2001; Kirby, 2008).

Users and Non-Users of Social Network Sites exhibit distinct social patterns (Hargittai, 2008) and construct their spaces (and times) in distinct manner (Zheng & Niu, 2007, Ahamer, 2010).

7.7 Design of social processes

Design can be performed in and of these following substrates :

- Time = theatre
- Space = architecture
- Geometry = graphics design
- Functionality = design in the narrower sense of industrial design)
- Interests = technology assessment (and administration)
- Structures = arts, science
- Love = new life.

In any of these cases it is necessary to en-act reality in and along time: time is the sequencer for all structures. Finally, en-acting means also the act of love: to “enact” life. Create life. New life.

Design is creative *generating* of structures. Hence it means the human share in continuous creation. Structures are created in an evolutionary way (Ahamer & Wahliss, 2008). On the other hand, science is (only) the *cognition* of structures.

Restrepo & Christiaans (2004: 3) find after several so-called ethnographic studies where they watch and analyse designers at work that “design is a *discursive* activity” i.e. a self-referenced process. “Designers propose design issues, reflect upon and discuss them and for each issue propose answers (also called positions). A discussion about which position to accept (design moves).” “However, problem structuring is not a clearly distinguishable phase of the design process but instead an activity that reoccurs regularly (...) and can contribute to either further structure the problem or to solve it.”

Thomas & Carroll (1979) discovered that designers tend to treat all problems as though they were ill-defined. They do so by changing the problems constraints and goals – even if the product was well-defined. Designers will be designers even if they can be problem solvers.

Dorst (2004) sees as the main “design problem” that “this process of reasoning is non-deductive”. There are “two ways in which a design problem is underdetermined:

1. a description in terms of needs, requirements and intentions can never be complete
2. ‘needs, requirements and intentions’ and ‘structure’ belong to different conceptual worlds.

He cites Dorst & Cross (2001) viewing creativity in the design process as a co-evolution of problem and solution spaces.

As a result, we suggest:

1. Rhythmisation of time (examples: theatre, evolution)
2. Rhythmisation in space and in opinion (examples: court, perspectives, politics, evolution of societal institutions, institution building).

Both these suggested strategies are implemented in the negotiation game “Surfing Global Change” (Ahamer, 2004).

Rhythmisation represents the theatrical strategy and *multi-perspectivism* represents the geo-locating strategy in the “space of perspectives”.

7.8 Space and time for consensus building

A suitable case study for a heuristic pattern for consensus building evolving in space and time is the five level web based negotiation game (Figure 17), its rules were published as (Ahamer, 2004).



Fig. 17. The „unfolder“ gives way to successively deeper levels of detail after embarking on reading (© G. Ahamer).

8. Transformation of coordinates

8.1 All the above examples have shown that

- various “spaces” can be thought of
- it would be suitable to enlarge the notion of “time”.

8.2 Suitably, a transformation of coordinates from time to “functional time” may be thought of.

8.3 In chapter 2, we suggested already to regard time as the substrate for procedures. Consequently, different “times” can be applied to different procedures. As an example, in theoretical physics, the notion of “Eigentime¹²” is common and means the system’s own time.

8.4 Similar to the fall line in the example of landslides in chapter 7.1 (red in Fig. 10) the direction of the functional time is the highest gradient of the envisaged process. This (any!) time axis is just a mental, cultural construction.

8.5 According to chapter 2 (Fig. 6) a clear understanding (mental model) is necessary to identify the main “effect of time”. We see that such an understanding can be culturally most diverse. Just consider the example of economic change:

- optimists think that the global income gap decreases with development
- pessimists believe that it increases, hampering global equity.

8.6 Therefore, any transformation of coordinates bears in itself the imponderability of complex social assumptions about future global development and includes a hypothesis on the global future.

8.7 Still, a very suitable transformation is

$$t \rightarrow \text{GDP/capita}$$

(Fig. 18) both because of good data availability and increased visibility of paths of development. GDP/cap resembles evolutionary time.

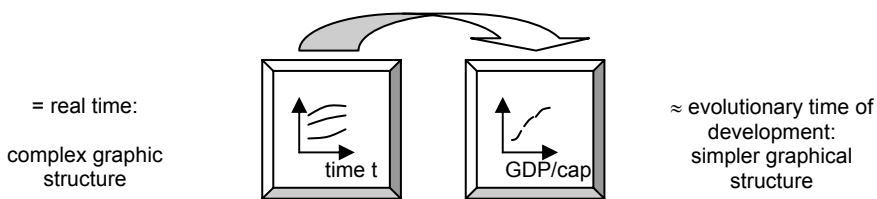


Fig. 18. A suitable transformation of time uses the economic level, measured as GDP per capita.

¹² literally (German): the own time (of the system)

8.8 The strategic interest of such a transformation is “pattern recognition”, namely to perceive more easily structures in data of development processes. Examples for such “paths of development” are shown in Fig. 19 for the example of fuel shares in energy economics.

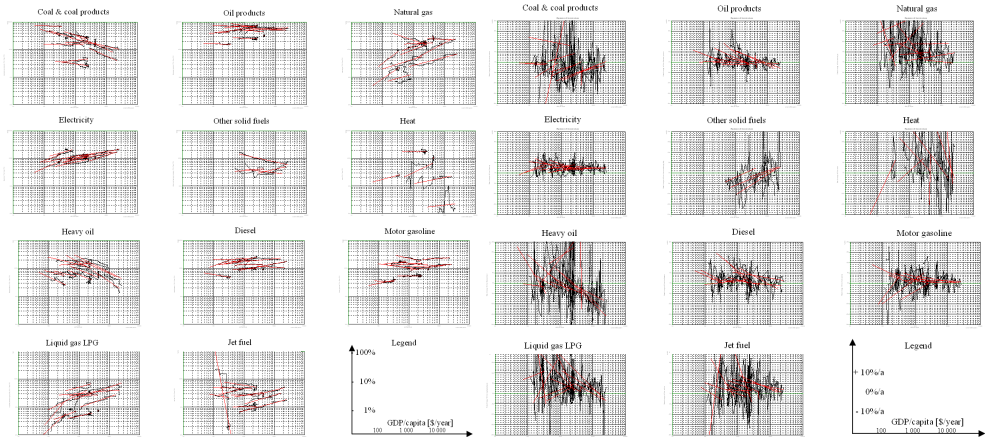


Fig. 19. Structural shift of percentages (left) and change rates of percentages (right) of various fuels in all nations' energy demand 1961-91. Data source: GCDB (Ahamer, 2001).

8.9 It is suggested here that implicitly during many mapping endeavours such transformation occurs. This is legitimate, but care must be taken to take into account the (silently) underlying model of human development.

8.10 Suitable transformation of coordinates can facilitate to see and communicate evolutionary structures, as it enables common views of humans and is therefore helpful for global consensus building.

8.11 Also the “effects of time” are projected into a common system of understanding which might give hope to facilitate common thinking independently of pre-conceived ideologies. This plan creates the “common reference system of objects”.

8.12 This paper suggests enlarging the concept of

- “globally universal geo-referencing” (one of the legacies of IT)

to

- “globally universal view-referencing”
- or “globally universal referencing of perspectives”¹³.

Fig. 20 illustrates this step symbolically.

¹³ The facts themselves may well be delivered by endeavours such as Wikipedia but here it refers to the perspective on facts! A huge voluntarily generated database on people's perceptions, views and opinions would be needed.

9. A futuristic vision

9.1 Building on the vision of “Digital Earth” (Gore, 1998), the deliberations in this paper might eventually lead to the vision of “Digital Awareness”: the common perspective on realities valid for the global population, aided by (geo)graphic means.

9.2 The primordial element of (human and societal) evolution is consensus building. Without ongoing creation of consensus global “evolutionary time” is likely to fall back.

The futuristic vision is to map global awareness.

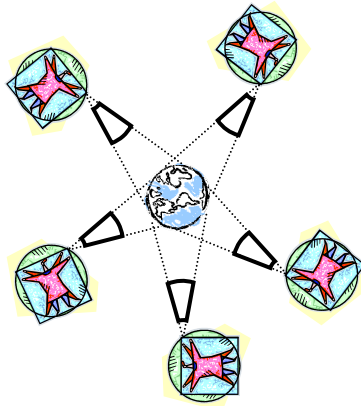


Fig. 20. The global society perceives the world.

9.3 Much like the georeferenced satellites which circulate around the world produce a “Google, Virtual [or similar] Earth”, the individual spectators in Fig. 20 circle around the facts – and they create a “common virtual perception”: an

IIS = Interperspective Information System.

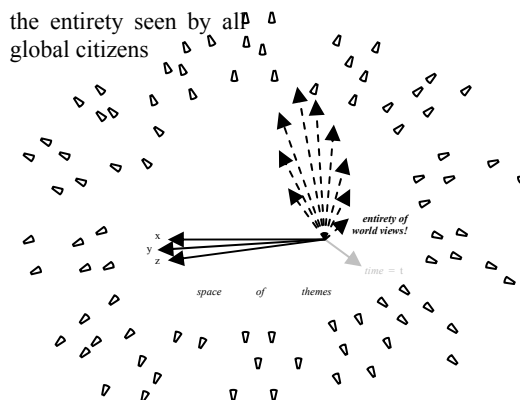


Fig. 21. Divergent perceptions circulate around earthen realities. The entirety of world views creates the IIS (Interperspective Information System).

9.4 Do we just mean interdisciplinarity? No. Nor do we simply refer to people looking into any direction. Fig. 22 shows the difference to IIS.

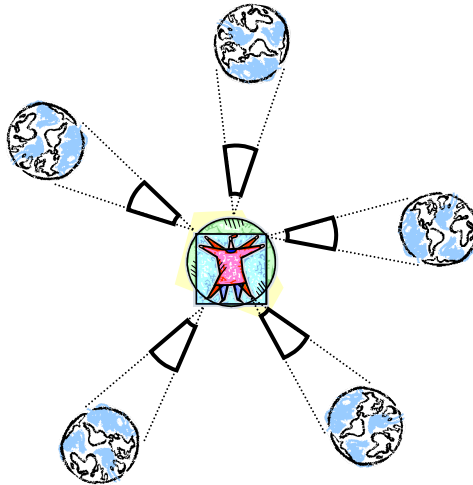


Fig. 22. This is not IIS.

9.5 The science of the third millennium will allow dealing with a multitude of world views and world perspectives (see Tab. 1) with an emphasis on consensus building. When learning, the emphasis lies on social learning and may also make use of game-based learning (such as the web-based negotiation game “Surfing Global Change”) which allows to experimentally experiment with world views without any risk involved.

	element • •	interaction • ↗ •	perspective ◁ ↗ •
single ones	Mechanics	Logics	Teaching
manifold	Thermo-dynamics	Systems analysis	Social learning gaming, IIS
	↓ 19 th cent.	↓ 20 th cent.	↓ 21 st cent.

Table 1. The science of the third millennium encompasses multiple perspectives

9.6 A suitable peaceful “common effort¹⁴” for a peaceful future of humankind would involve developing tools and visual aids in order to understand the opinions of other citizens of the globe.

The future is dialogue.

Or else there will be no future.

¹⁴ جهاد (jihad in Arabic) also means: common effort of a society

10. Conclusion

Sciences are similar to “languages” spoken by people, they differ globally. Understanding for others’ languages is essential for global sustainable peace.

Human perceptions are also strongly influenced by underlying models, assumptions and preconceived understandings.

Studying geo-referenced data sets (GIS) can help to facilitate bridging interperceptual gaps.

For the transformation of world views – to make them understandable – it is necessary to know about

- the “effect of time”, namely the “path along the continuum of time” which a variable is expected to take
- the speakers’ underlying model of a complex techno-socio-economic nature
- the resulting perception of other humans.

A future task and purpose of IT could be to combine the multitude of (e.g. geo-referenced) data and to rearrange it in an easily understandable manner for the viewpoints and perspectives of another scientific discipline or just another human being. Such a system is called Interperspective Information System IIS.

Merging a multitude of perspectives to form a common view of the entire global population is the target of an IIS.

Symbolically, a “Google Earth”-like tool would eventually develop into a “Google World Perspective”-like tool, or a “Virtual Earth”-like tool would become a “Virtual Perspective” tool encompassing all (scientific, social, personal, political, etc.) views in an easily and graphically understandable manner.

In the above futuristic vision, IT can/should(!) become a tool to facilitate consensus finding. It can rearrange the same data for a new view.

Symbolically speaking: similar to Google Earth which allows one to view the same landscape from different angles, a future tool would help to navigate the world concepts, the world views and the world perspectives of the global population.

IT can reorganise extremely large data volumes (if technological growth rates continue) and could eventually share these according to the viewpoint of the viewer.

Such a step of generalisation would lead from “Geographic Information Science” to “Interperspective Information Science”, implying the change of angles of perception according to one’s own discipline.

Dialogue represents the ultimate heuristics for complex interdisciplinary and intercultural problems. Science does not offer more than such dialogue.

11. References

- Ahamer, G. (2001). A Structured Basket of Models for Global Change. In: *Environmental Information Systems in Industry and Public Administration (EnvIS)*. ed. by C. Rautenstrauch and S. Patig, Idea Group Publishing, Hershey, 101-136, http://www.oeaw-giscience.org/ProjectFactSheets/ProjectFactSheet_GlobalChange.pdf.
- Ahamer, G., Wahlliss, W. (2008). *Baseline Scenarios for the Water Framework Directive*. Ljubljana, WFD Twinning Project in Slovenia, http://www.oeaw-giscience.org/ProjectFactSheets/ProjectFactSheet_EU_SDI.pdf.
- Ahamer, G., & Strobl, J. (2009). Learning across social spaces. In: *Cases on Technological Adaptability and Transnational Learning*, IGI Publishing, Hershey, USA, pp. 1-24.
- Ahamer, G. (2010). Heuristics of social process design, pp. 265-298.
- Andrienko, N., Andrienko G. (2006). *Exploratory Spatial Analysis*, Springer
- Brunner, F.K., Zobl, F., Gassner, G. (2003). On the Capability of GPS for Landslide Monitoring. *Felsbau* 2/2003, 51-54.
- Castells, M. (1998). *The Information Age: Economy, Society and Culture*. Trilogy containing three volumes. Cambridge, MA; Oxford, UK: Blackwell.
- Crang, M. (2000). Public Space, Urban Space and Electronic Space: Would the Real City Please Stand Up? *Urban Studies*, 37(2), 301-317.
- de Chardin, T. (1950). *La condition humaine* [Der Mensch im Kosmos]. Beck, Stuttgart.
- Dodge, M., McDerby, M., Turner, M. (eds.) (2008) *Geographic Visualisation*, Wiley
- Dorst, C.H. & Cross, N.G. (2001). Creativity in the design process: co-evolution of problem-solution, *Design Studies*, 22, 425-437.
- Dorst, K. (2004). On the Problem of Design Problems: problem solving and design expertise. *Journal of Design Research*, 4(2).
- Dykes, J., A. MacEachren, et al. (2005). *Exploring Geovisualization*. Oxford, Elsevier.
- Fischer F. (2008a): Location Based Social Media - Considering the Impact of Sharing Geographic Information on Individual Spatial Experience. In: Car A., Griesebner G. a. J.Strobl (Eds.): *Geospatial Crossroads @ GI_Forum '08*. Proceedings of the Geoinformatics Forum Salzburg, pp. 90-96.
- Fischer F. (2008b): Implications of the usage of mobile collaborative mapping systems for the sense of place. In: M. Schrenk. et al. (Eds.): *REAL CORP 008: Mobility Nodes as Innovation Hubs*. Proceedings of 13th International Conference on Urban Planning, Regional Development and Information Society, pp. 583-587.
- Fischer F. (2008c): Microsoft Virtual Earth. Integrating Geospatial Technology in Everyday Life. *Geoinformatics*, 4(11), 6-9.
- Galileo, G. (1632). *Dialogo sopra i due massimi sistemi del mondo, tolemaico, e copernicano*. Fiorenza.
- GIScience, (2008). Connecting Real and Virtual Worlds. Poster at AGIT'08, http://www.oeaw-giscience.org/index.php?option=com_content&task=blogcategory&id=43&Itemid=29.
- Gore, A. (1998). Vision of Digital Earth, http://www.isde5.org/al_gore_speech.htm.
- Grimm, V. & S. F. Railsback, Eds. (2005). *Individual-based Modeling and Ecology*. Princeton Series in Theoretical and Computational Biology. Princeton University Press, Princeton and Oxford.

- Hargittai, E. (2008). Whose Space: Differences Among Users and Non-Users of Social Network Sites. *Journal of Computer-Mediated Communication*, 13(1), 276–297.
- Kirby, A. (2008). The production of private space and its implications for urban social relations. *Political Geography*, 27(1), 74-95
- Klima, K. & Zobl F. (2009, in press). Herausforderung der geologischen Erkundung und der Untergrundmodellierung für die geotechnische Analyse Herausforderung der geologischen Erkundung und der Untergrundmodellierung für die geotechnische Analyse. Key note paper published in: *Tagungsband Computerorientierte Geologie 2009* in Salzburg, Wichmann.
- Kraak (2009). Minard's map. www.itc.nl/personal/kraak/1812/3dnap.swf
- Longley, P.A. et al. (2001) *Geographic Information. Science and Systems*. Wiley
- MacEachren, A. M., M. Gahegan, et al. (2004). Geovisualization for Knowledge Construction and Decision Support. *IEEE Computer Graphics & Applications* 2004 (1/2): 13-17.
- Marschallinger, R. (2009). Analysis and Integration of Geo-Data. <http://www.oeaw-giscience.org/>.
- Peuquet, D. J. (2002). *Representations of Space and Time*. New York, The Guilford Press.
- Restrepo, J., Christiaans, H. (2004). Problem Structuring and Information Access in Design. *Journal of Design Research*, 4(2).
- Thomas, J. & Carroll, J. (1979). The psychological study of design. *Design Studies*, 1(1), 5-11.
- Wallentin, G., U. Tappeiner, J. Strobl & E. Tasser (2008). Alpine tree line dynamics: an individual based model. *Ecological Modelling*, 218(3-4). p. 235-246, doi:10.1016/j.ecolmodel.2008.07.005.
- Wallentin, G. (2009). Ecology & GIS. Spatiotemporal modelling of reforestation processes. See <http://www.oeaw-giscience.org/images/stories/Downloads/pecha%20kucha%20technoz%20day.pdf>
- Zheng, J. & Niu, J. (2007). Unified Mapping of Social Networks into 3D Space. IMSCCS 2007, Second International Multi-Symposiums on Computer and Computational Sciences. See <http://ieeexplore.ieee.org/xpl/conferences.jsp>.
- Zobl, F. (2009). Mapping, Modelling and Visualisation of georelevant processes. <http://www.oeaw-giscience.org/>.

Complexity of Instances for Combinatorial Optimization Problems

Jorge A. Ruiz-Vanoye¹, Ocotlán Díaz-Parra¹,
 Joaquín Pérez-Ortega², Rodolfo A. Pazos R.³
 Gerardo Reyes Salgado² and Juan Javier González-Barbosa³

¹ *Universidad Autónoma del Estado de Morelos, Mexico*

² *Centro Nacional de Investigación y Desarrollo Tecnológico, Mexico*

³ *Instituto Tecnológico de Ciudad Madero, Mexico*

1. Introduction

The theory of the computational complexity is the part of the theory of the computation that studies the resources required during the calculation to solve a problem (Cook, 1983). The resources commonly studied are the time (execution number of an algorithm to solve a problem) and the space (amount of resources to solve a problem). In this area exist problems classifications that are approached within the theory of the complexity, some definitions of the complexity classes are related to this investigation:

- a. P class.-It is the class of recognizable languages by a determinist Turing Machine of one tape in polynomial time (Karp, 1972).
- b. NP class.-It is the class of recognizable languages by a Non-determinist Turing Machine of one tape in polynomial time (Karp, 1972).
- c. NP-equivalent class.-It is the class of problems that are considered NP-easy and NP-hard (Jonsson & Bäckström, 1995).
- d. NP-easy class.-It is the class of problems that are recognizable in polynomial time by a Turing Machine with one Oracle (subroutine). In other words a problem X is NP-easy if and only if a Y problem exists in NP like X is reducible Turing in polynomial (Jonsson & Bäckström, 1995).
- e. NP-hard class.-A Q problem is NP-hard if each problem in NP is reducible to Q (Garey & Johnson, 1979; Papadimitriou & Steiglitz, 1982). It is the class of problems classified as problems of combinatorial optimization at least as complex as NP.
- f. NP-complete class.-A L language is NP-complete if L is in NP, and Satisfiability $\leq_p L$ (Cormen et al., 2001; Karp, 1972; Cook, 1971). It is the class of problems classified like decision problems.

A combinatorial optimization problem is either a minimization problem or a maximization problem and consists of three parts: a) a set of instances, b) candidate solutions for each instance, c) a solution value (Garey & Johnson, 1979). The combinatorial optimization problems that was used in this paper: the General Asymmetric Traveling Salesman Problem

(ATSP), JobShop Scheduling Problem (JSSP) and Vehicle Routing Problem with Time Windows (VRPTW).

The general Asymmetric Traveling Salesman Problem (ATSP), which can be stated as follows: given a set of nodes and distances for each pair of nodes, find a route of minimal overall length that visits each of the nodes exactly once (Cormen et al., 2001). The distance of node i to node j and the distance of node j to node i can be different (equations 1, 2, 3, 4).

$$\min z(x) = \sum_{j=1}^m \sum_{i=1}^m d_{ij} x_{ij} \quad (1)$$

$$\sum_{j=1}^m x_{ij} = 1; \quad i = 1, \dots, m; \quad d_{ij} \neq d_{ji} \quad (2)$$

$$\sum_{i=1}^m x_{ij} = 1; \quad j = 1, \dots, m; \quad 0 \leq x_{ij} \leq 1 \quad (3)$$

$$x_{ij} = \begin{cases} 0, & \text{if tour traverses from } i \text{ to } j \\ 1, & \text{otherwise} \end{cases} \quad \forall i, j \quad (4)$$

The JobShop Scheduling Problem (JSSP) contains a number of machines and a set of Jobs each one with precedence restrictions, the problem is to solve the question if exist a scheduling of jobs that help to improve and to efficiency the use of the machines being eliminated the idle times. It is recognized by that it does not have to be able human nor machine sufficiently fast that it can obtain the optimal solution for JSSP due to the solutions space, which cannot be expressed by a polynomial function (deterministic algorithm), the space of solutions for this kind of problem can be only expressed like an exponential function. For the problem of JSSP is necessary to diminish makespan (c_{max}), this can be formulated as it follows (equations 5, 6, 7, 8):

$$\min c_{\max} \quad (5)$$

$$c_{jk} - t_{jk} \geq c_{jh}, \quad j = 1, 2, \dots, n \quad h, k = 1, 2, \dots, m \quad (6)$$

$$c_{jk} - c_{ik} + M(1 - x_{ijk}) \geq t_{jk}, \quad i, j = 1, 2, \dots, n \quad k = 1, 2, \dots, m \quad (7)$$

$$c_{ik}, c_{jk} \geq 0, \quad x_{ijk} = 1 \text{ or } 0, \quad i = 1, 2, \dots, n \quad k = 1, 2, \dots, m \quad (8)$$

The Vehicle Routing Problem with Time Windows (VRPTW) is a combinatorial optimization problem complex (Toth & Vigo, 2001; Ruiz-Vanoye et al., 2008b; Cruz-Chávez et al., 2008). The VRPTW (Toth & Vigo, 2001) consists basically of to minimize the costs of subject transportation to time restrictions of each route and capacity on the cradle of the demand of each client (equations 9 and 10).

$$\min \sum_{k \in K} \sum_{(i,j) \in A} c_{ij} x_{ijk} \quad (9)$$

$$a_i \sum_{j \in \Delta^+(i)} x_{ijk} \leq w_{ik} \leq b_i \sum_{j \in \Delta^+(i)} x_{ijk}, \quad \forall k \in K, i \in N \quad (10)$$

In this paper, we propose specify the complexity of instances for problems of combinatorial optimization (ATSP, VRPTW, and JSSP).

2. Related works

An important measurement of complexity that we can attribute to Shannon (1949), is the complexity of Boolean circuits. For this measurement he is advisable to assume that f function at issue transforms finite strings of bits into finite strings of bits, and the complexity of f is the size of the smaller Boolean circuit than it calculates f for all the inputs of n length. But, it does not exist a classification of the complexity of instances for combinatorial optimization problems.

In some combinatorial optimization problems exists diverse types of instances size at the moment, for example:

REINELT (Reinelt, 1991) mentions that in General Asymmetric Traveling Salesman Problem (ATSP) instances exist whose size based on the number of cities or nc . See Table 1.

YAMADA (Yamada & Nakano, 1997) mentions that for the problem JobShop Scheduling Problem (JSSP) the size of the problem is the number of jobs or J and the number of machines or M . See Table 2.

SOLOMON (Solomon, 1987; Toth & Vigo, 2001) mentions that in the problem Vehicle Routing Problem with Time Windows (VRPTW) exists classifications of type C for instances clustered, type RC for instances Random and Clustered, Type R for Random instances and in addition the instance is determining by the number of clients or CN . See Table 3.

<i>Instances</i>	<i>nc</i>
br17	17
ft53	53
ft70	70
ftv33	33
ftv35	35
ftv38	38
ftv55	55
ftv64	64
ftv70	70
ftv170	170
rbg443	443
ry48p	48

Table 1. Number of cities for ATSP instances

<i>Instances</i>	<i>J*M</i>
abz5	10x10
abz6	10x10
abz8	20x15
ft06	6x6
ft10	10x10
ft20	20x5
orb01	10x10
orb02	10x10
yn1	20x20
yn2	20x20
yn3	20x20
yn4	20x20

Table 2. Number of jobs and number of machines for JSSP instances

<i>Instances</i>	<i>CN</i>
c101	25
c102	25
c205	25
r101	50
r102	50
r201	100
r205	100
rc101	200
rc102	200
rc201	500
rc202	500
rc203	500
rc204	500
rc205	500

Table 3. Client number for VRPTW instances

3. Complexity of Instances for Combinatorial Optimization Problems

In this section, we propose a basic methodology (Fig. 1) for create the metric that permit to classify or to determine the complexity instances of combinatorial optimization problems.

Step 1. To identify the maximum instance solved of the problem.

Step 2. To identify the instances parameters of the problem.

Step 3. Elaboration of the general metric of instances complexity taking into account measured of descriptive statistics and the parameters of the problems (equation 11).

$$InstancesComplexity = (PS + AM + SD + GD + RA) / 100 \tag{11}$$

Where: *PS* is the instance average size at the rate of the instance maximum solved of the problem, *AM* is the sum of the arithmetic mean of each instance parameters (that have several elements), *SD* is the sum of the standard deviations of each type of parameter (that have several elements), *GD* is the sum of the genetic distances which quantifies the similarity or differentiates between the populations from the frequencies of the elements from each parameters (that have several elements) of the problem, *RA* is the sum of the existing reasons between the greatest value and the value smaller of each parameter (that has several elements) of the problem.

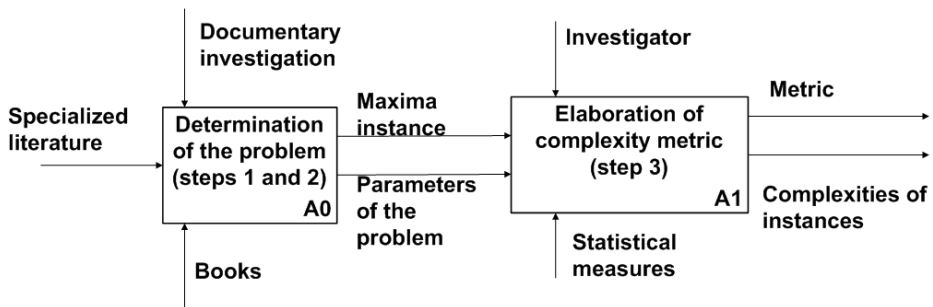


Fig. 1. Methodology of complexity instances for computational combinatorial problems

4. Experimentation

The experimentation was performed on a computer with Celeron processor 1.5GHz, 512 MB of memory, 80 GB of hard disk.

We use: the ATSP instances obtained of TSPLIB (Reinelt, 1991) benchmark, the JSSP instances of JSSPLIB (Yamada & Nakano, 1997) benchmark, the VRPTW instances of Solomon (Solomon, 1987) benchmark, and a genetic algorithm from Heuristics Lab (Wagner & Affenzeller, 2004) software.

A genetic algorithm (GA) is one of the heuristic methods used to find approximate solutions to NP-complete problems, GA is inspired by the Darwinian principles of the evolution of the species, and use own techniques of the genetics, such as: inheritance, mutation, natural selection and recombination (or crossover). The simplest form of genetic algorithm involves three types of operators: selection, crossover (single point), and mutation (Holland, 1975; Mitchell, 1998). Selection, this operator selects chromosomes in the population for reproduction. The fitter the chromosome, the more times it is likely to be selected to reproduce. Crossover, this operator randomly chooses a locus and exchanges the subsequences before and after that locus between two chromosomes to create two offspring. The crossover operator roughly mimics biological recombination between two single-chromosome organisms. Mutation, this operator randomly flips some of the bits in a chromosome. Mutation can occur at each bit position in a string with some probability. The genetic algorithm (Wagner & Affenzeller, 2004) was used to verify the time and the quality of instances solution with the purpose of determining if the metric generated classify in complexity terms. The input parameters were: selection operator = Roulette, crossover operator = OX, mutation operator = Simple Inversion, generations = 1000, population size = 100, mutation rate = 0.05, replacement strategy = Elitism, crossover rate = 1, n-Elitism = 1, tournament group size = 2.

4.1 Experimentation in ATSP

Using the process contained in the general methodology to ATSP, the steps they would be of the following way:

Step 1. The maximum instance solved for ATSP is of 1.904.711 cities or World TSP (Applegate et al., 2006).

Step 2. The instances parameters of general ATSP can be codified in a hypothetical language: $L = "nc, d_1, d_2, \dots, d_z"$. In other words two general parameters nc = number of cities and d_z =distances between the cities.

Step 3. Elaboration of the instances complexity metric for ATSP. In order to create the complexity general metric, for which it is necessary to determine PS (equation 12), AM (equation 13), SD (equation 14), GD (equation 15), RA (equation 16) indicators.

$$PS = \frac{nc}{nc_{max}} \quad (12)$$

Where: PS is the average size of the instance at the rate of the instance maximum solved of ATSP, nc is the value of the problem to solve and nc_{max} is the size of greater instance solved.

$$AM = \sum_{i=1}^n \frac{d_i}{nc} \quad (13)$$

Where: AM is the sum of the arithmetic mean of each parameter (that has several elements) of the instance, d_i are the elements of the parameter that has several elements called matrix of cities and nc is the number of cities.

$$SD = \sqrt{\left(\sum_{i=1}^n \left(AM - \left(\frac{d_i}{nc} \right) \right)^2 \right)} \quad (14)$$

Where: SD is the sum of the standard deviations of each type of parameter (that has several elements), d_i = Value of distances between cities i , nc = number of cities, AM = index of the arithmetic mean of distances.

$$GD = \begin{cases} \sum \text{Frequency}(d_i) > 1, & \text{if } \text{Frequency}(d_i) > 1 \text{ for some } i \\ 0, & \text{otherwise} \end{cases} \quad (15)$$

Where: GD is the sum of the genetic distances which quantifies the similarity or differentiates between the populations from the frequencies of the elements from each parameter of the problem.

$$RA = \frac{\min d_i}{\max d_i} \quad (16)$$

Where: RA is the sum of the existing reasons between the greatest value and the smallest value of each parameter (that has several elements) of the NP-hard problem, $\min d_i$ = minimal distance between cities, $\max d_i$ = value of maximum distances between cities.

In Table 4 are the results of applying the instances complexity metric (IC) on ATSP instances, in addition to the summary of the obtained result to execute algorithm GA with the ATSP instances.

Instances	nc	GA		PS	AM	SD	RA	GD	IC
		d	t						
br17	17	0	0.79	0.000008	12.12	18.45	0.16	0.53	0.31
ft53	53	45.2	1.72	0.000027	283.15	377.02	0.15	0.32	6.60
ft70	70	28.8	3.68	0.000036	721.74	434.82	0.27	0.24	11.57
ftv33	33	42	3.21	0.000017	88.47	61.93	0.26	0.22	1.50
ftv35	35	26.6	3.22	0.000018	139.07	64.33	0.41	0.36	2.04
ftv38	38	38.1	3.09	0.000019	68.14	62.52	0.20	0.26	1.31
ftv44	44	56.3	3.17	0.000023	70.18	63.31	0.21	0.30	1.34
ftv47	47	43.8	3.09	0.000024	145.40	65.45	0.41	0.48	2.11
ftv55	55	94.0	3.16	0.000028	88.99	61.52	0.28	0.56	1.51
ftv64	64	75.9	2.91	0.000033	21.83	68.26	0.06	0.65	0.90
ftv70	70	88.1	3.61	0.000036	22.39	69.10	0.07	0.71	0.92
ftv170	170	325	3.91	0.000089	77.18	69.30	0.21	1.70	1.48
rbg443	443	124	6.33	0.000232	1.77	8.27	0.03	894.00	9.04
ry48p	48	1067	3.04	0.000025	264.11	578.73	0.96	0.12	8.43

Table 4. Results obtained from descriptive statistical measures with the ATSP instances

Where: d is the difference between best know and the obtained solution, t is the solution time for the instance on the GA algorithm. In the instance complexity (IC) between $ft70$ and $ftv70$ instances exist different values for instances with equal number of cities (nc), indicates that the $ft70$ instance is more complex than $ftv70$ instance and is verified with the obtained solution to apply the GA algorithm to the instances. In addition to being able to compare between the $ftv70$ and $ftv64$ instances, the $ftv70$ instance is more complex than $ftv64$ instance.

4.2 Experimentation in JSSP

Using the methodology with problem JSSP, are the following steps:

Step 1. The maximum instance resolved is of 20 by 20 symmetrical (Yamada & Nakano, 1997).

Step 2. The parameters of the instances of the JSSP can be codified in a hypothetical language: $L = "nj, nm, J_1(m_0, pt_0, \dots, m_{nm-1}, pt_{nm-1}), \dots, J_{nj}(m_0, pt_0, \dots, m_{nm-1}, pt_{nm-1})$. Where nj = Number of jobs, nm = number of machines, J = Jobs, m = machine and pt = processing time.

Step 3. Elaboration of the complexity metric for JSSP. In order to create the complexity general metric is used the equation 1, for which it is necessary to determine PS (equation 17), AM (equation 18), SD (equation 19), DG (equation 20), RA (equation 21) indicators.

$$PS = \left(\frac{nj * nm}{MAXJSSP} \right) \quad (17)$$

Where: PS is the so large average of the instance at the rate of the maximum instance from problem JSSP. And $MAXJSSP$ = the maximum instance solved in literature or $MAX(nj * nm)$.

$$AM = \sum_{i=1}^z \frac{m_i}{m_{(nj*nm)}} + \sum_{i=1}^z \frac{pt_i}{pt_{(nj*nm)}} \quad (18)$$

Where the AM metric is the sum of the arithmetic mean of the m, pt parameters.

$$SD = \sqrt{\left(\sum_{i=1}^n \left(AMm - \left(\frac{m_i}{m_{(nj*nm)}} \right) \right) \right)^2} + \sqrt{\left(\sum_{i=1}^n \left(AMpt - \left(\frac{pt_i}{pt_{(nj*nm)}} \right) \right) \right)^2} \quad (19)$$

Where the SD metric contains the sum of the standard deviations of the m, pt, J parameters. And MA is the arithmetic mean of m parameter; $MApt$ is the arithmetic mean of pt parameter.

$$GD = [Frequency(m) + Frequency(pt)] / 100 \quad (20)$$

Where the GD metric contains the genetic distances between the populations by the frequencies of the parameters (m, pt).

$$RA = \frac{MIN m}{MAX m} + \frac{MIN pt}{MAX pt} \quad (21)$$

Where the RA metric contains the sum of the reasons between the minimal and maximum values of the m and pt parameters.

The Table 5 contains the results of applied the complexity metric for JSSP, and the results of the GA on JSSP instances. This can be observed that in the Instance Complexity (IC) between la01 and la05 instances exist different values with equal Number of Jobs and Number of Machines, indicates that the la01 instance is more complex than the la05 instance is verified with the quality of the obtained solution. Also sample that in between the yn1 and yn4 instances, the yn4 instance is more complex that yn1 instance. Where: d is the difference between best know and the obtained solution, t is the solution time for the instance on the GA algorithm.

Instances	J*M	GA		TP	MA	DA	RA	DG	IC
		d	t						
abz5	10x10	4.29	0:55.3	0.25	73.64	2302.69	0.505	0.10	23.77
abz6	10x10	3.39	1:09.3	0.25	58.51	1830.02	0.204	0.10	18.89
abz8	20x15	16.6	2:00.8	0.75	29.12	2247.37	0.275	0.20	22.77
abz9	20x15	14.3	1:46.5	0.75	29.28	1862.31	0.275	0.02	18.92
ft06	6x6	0	0:08.5	0.09	7.25	156.57	0.100	0.06	1.64
ft10	10x10	8.06	0:28.4	0.25	51.64	1615.42	0.020	0.10	16.67
ft20	20x5	3.43	0:35.3	0.25	51.16	2280.33	0.020	0.20	23.31
la01	10x5	0	0:13.7	0.12	53.82	1177.07	0.122	0.10	12.31
la02	10x5	0.15	0:15.2	0.12	50.24	1098.87	0.121	0.10	11.49
la03	10x5	4.35	0:04.3	0.12	44.22	966.58	0.076	0.10	10.11
la04	10x5	2.03	0:14.5	0.12	46.22	1010.19	0.051	0.10	10.56
la05	10x5	0	0:13.5	0.12	40.06	874.45	0.051	0.10	9.14
la06	15x5	0	0:22.3	0.18	51.78	1730.97	0.071	0.10	17.83
la07	15x5	0	0:25.4	0.18	48.13	1608.84	0.082	0.10	16.57
la08	15x5	0	0:25.9	0.18	49.05	1639.64	0.051	0.10	16.88
la09	15x5	0	0:24.6	0.18	54.62	1825.91	0.070	0.10	18.80
la10	15x5	0	0:23.1	0.18	53.28	1781.12	0.051	0.10	18.34
orb01	10x10	5.00	0:28.7	0.25	53.36	1668.86	0.050	0.10	17.22
orb02	10x10	5.96	0:23.4	0.25	50.91	1591.87	0.060	0.10	16.43
orb03	10x10	9.65	0:24.4	0.25	52.79	1651.21	0.050	0.10	17.04
yn1	20x20	0.78	1:15.8	1.00	36.12	3217.55	0.204	0.25	32.55
yn2	20x20	2.78	1:18.1	1.00	36.23	3227.61	0.204	0.26	32.65
yn3	20x20	2.07	1:19.0	1.00	32.07	3169.99	0.204	0.27	32.07
yn4	20x20	0.80	1:25.8	1.00	36.84	3281.54	0.204	0.31	33.19

Table 5. Results obtained by the descriptive statistics with the JSSP instances

4.3 Experimentation in VRPTW

Using the methodology with VRPTW, the following steps are obtained:

Step 1. The maximum instance solved of problem VRP is F-n135-k7, with $VN = 12$, $C = 2210$, $CN = 135$ (Toth & Vigo, 2001).

Step 2. The parameters of the instances of problem VRPTW can be codified in a hypothetical language: $L = "VN, C, (CN_1, XCO_1, YCO_1, D_1, RT_1, DT_1, ST_1, \dots, CN_z, XCO_z, YCO_z, D_z, RT_z, DT_z, ST_z)"$. Where VN = Vehicle Number, C = Capacity, CN = Customer Number, XCO = X Coord., YCO = Y Coord., D = Demand, RT = Ready Time, DT = Due date, ST = Service Time.

Step 3: Elaboration of the complexity metric for problem VRPTW. In order to create the complexity general metric the equation 1 is used, for which it is necessary to determine PS (equation 22), AM (equation 23), SD (equation 24), GD (equation 25), RA (equation 26) indicators.

$$PS = \frac{VN * C * CN_n}{VRPTWMAX}, VRPTWMAX = MAX(VN * C * CN_n) \quad (22)$$

Where: PS = is the problem size of the instance at the rate of the maximum instance solved from VRPTW, $\max(VN * C * CN_n)$ = the value of the maximum instance solved by literature.

$$AM = \sum_{i=1}^n \frac{XCO_i}{CN_z} + \sum_{i=1}^n \frac{YCO_i}{CN_z} + \sum_{i=1}^n \frac{D_i}{CN_z} + \sum_{i=1}^n \frac{RT_i}{CN_z} + \sum_{i=1}^n \frac{DT_i}{CN_z} + \sum_{i=1}^n \frac{ST_i}{CN_z} \quad (23)$$

Where: In AM contains the sum of the arithmetic mean of the XCO , YCO , D , RT , DT y ST parameters.

$$SD = \sqrt{\sum_{i=1}^n \left(AMXCO - \left(\frac{XCO_i}{CN_z} \right) \right)^2} + \sqrt{\sum_{i=1}^n \left(AMYCO - \left(\frac{YCO_i}{CN_z} \right) \right)^2} + \sqrt{\sum_{i=1}^n \left(AMD - \left(\frac{D_i}{CN_z} \right) \right)^2} + \sqrt{\sum_{i=1}^n \left(AMRT - \left(\frac{RT_i}{CN_z} \right) \right)^2} + \sqrt{\sum_{i=1}^n \left(AMDT - \left(\frac{DT_i}{CN_z} \right) \right)^2} + \sqrt{\sum_{i=1}^n \left(AMST - \left(\frac{ST_i}{CN_z} \right) \right)^2} \quad (24)$$

Where: SD contains the sum of the standard deviations of the XCO , YCO , D , RT , DT y ST parameters. $AMXCO$ is the arithmetic mean of XCO parameter, $AMYCO$ is the arithmetic mean of YCO parameter, AMD is the arithmetic mean of D parameter, $AMRT$ is the arithmetic mean of RT parameter, $AMDT$ is the arithmetic mean of DT parameter, and $AMST$ is the arithmetic mean of ST parameter.

$$GD = [Frequency(XCO) + Frequency(YCO) + Frequency(D) + Frequency(RT) + Frequency(DT) + Frequency(ST)] / 100 \quad (25)$$

Where: GD contains the genetic distances between populations from the frequencies of each parameter (XCO , YCO , D , RT , DT y ST).

$$RA = \left(\frac{MIN XCO}{MAX XCO} \right) + \left(\frac{MIN YCO}{MAX YCO} \right) + \left(\frac{MIN D}{MAX D} \right) + \left(\frac{MIN RT}{MAX RT} \right) + \left(\frac{MIN DT}{MAX DT} \right) + \left(\frac{MIN ST}{MAX ST} \right) \quad (26)$$

Where RA contains the sum of the reasons between the minimal and maximum values of the XCO , YCO , D , RT , DT y ST parameters.

In Table 6 are the results of applying the complexity metric on VRPTW instances, in addition to the summary of the obtained result to execute algorithm GA with the VRPTW instances. Where: d is the difference between best know and the obtained solution, t is the solution time for the instance on the GA algorithm, VN = Vehicle Number, C = Capacity, CN = Customer Number. It can be observer that in the Instance Complexity (IC) between

c204 and c201 different values for instances with equal Vehicle Number Capacity and Customer Number exist, this indicates that the instance c204 is more complex than c201 and is verified with the obtained solution to apply GA to the instances.

It is necessary to mention that the values of complexity of the instances of single VRPTW are comparable between instances of the same problem.

<i>Instances</i>	VN	C	C N	GA		TP	MA	DA	RA	DG	IC
				d	t						
c101	25	200	25	0	0:58	0.034	1143.92	28598	1.017	0.10	297.43
c102	25	200	25	0.05	1:03	0.034	1158.68	28967	1.022	0.17	301.26
c103	25	200	25	0	1:00	0.034	1284.92	32123	1.037	0.23	334.09
c104	25	200	25	0	1:01	0.034	1303.68	32592	1.037	0.27	338.97
c105	25	200	25	0	1:26	0.034	1151.12	28778	1.060	0.10	299.30
c201	25	1000	25	0	1:35	0.122	3657.96	91449	0.894	0.08	951.08
c202	25	700	25	0	1:34	0.122	3898.96	97474	0.894	0.15	1013.74
c203	25	700	25	0	2:03	0.122	4032.08	100802	0.897	0.21	1048.35
c204	25	700	25	0.1	1:48	0.122	4081.20	102030	1.263	0.25	1061.12
c205	25	700	25	0	1:44	0.122	3675.56	91889	0.941	0.08	955.65
r101	25	200	25	0.06	1:40	0.034	310.36	7759	0.431	0.09	80.69
r102	25	200	25	1.37	1:11	0.034	318.44	7961	0.431	0.16	82.80
r103	25	200	25	1.12	1:19	0.034	317.96	7949	0.558	0.22	82.67
r104	25	200	25	0.16	1:54	0.034	309.56	7739	0.558	0.26	80.49
r105	25	200	25	2.15	1:20	0.034	310.68	7767	0.484	0.09	80.78
r201	25	700	25	1.19	1:49	0.174	1058.32	26458	0.458	0.09	275.17
r202	25	700	25	0.39	2:09	0.174	1122.44	28061	0.492	0.16	291.84
r203	25	700	25	0.83	2:05	0.174	1146.04	28651	0.549	0.22	297.97
r204	25	700	25	2.66	2:04	0.174	1114.84	27871	0.549	0.26	289.86
r205	25	700	25	0	2:00	0.174	1057.80	26445	0.509	0.09	275.03
rc101	25	700	25	0	1:22	0.034	326.72	8168	0.329	0.08	84.95
rc102	25	700	25	0	1:17	0.034	327.72	8193	0.329	0.15	85.21
rc103	25	700	25	0	2:09	0.034	323.36	8084	0.425	0.21	84.08
rc104	25	700	25	0.11	1:27	0.034	313.28	7832	0.425	0.25	81.45
rc105	25	700	25	0.44	1:15	0.034	335.04	8376	0.383	0.08	87.11
rc201	25	700	25	0	1:42	0.174	1024.00	25600	0.220	0.08	266.24
rc202	25	700	25	0	2:03	0.174	1077.24	26931	0.220	0.15	280.08
rc203	25	700	25	0.3	2:38	0.174	1099.20	27480	0.365	0.21	285.79
rc204	25	700	25	0	2:07	0.174	1067.52	26688	0.365	0.25	277.56
rc205	25	700	25	0	2:14	0.174	1044.88	26122	0.333	0.08	271.67

Table 6. Results obtained by metrics descriptive statistics with VRPTW instances

5. Conclusions

We can conclude that the creation of a mathematical expression based on the descriptive statistics able is possible to measure the complexity the instances of combinatorial optimization problems, in this paper was demonstrated for ATSP, VRPTW and JSSP. It is necessary to mention that, the values of instances complexity of ATSP, VRPTW and JSSP problems are only comparable between instances of the same problem.

The intention of this paper was to classify the complexity of the instances and not therefore the complexity between problems NP, as future works, we propose to validate the methodology for other NPs problems.

6. References

- Applegate, D.L.; Bixby, R.E.; Chvátal Y. & Cook, W.J. (2006). *The Traveling Salesman Problem: A Computational Study*, Princeton University Press, ISBN:0691129932, New Jersey
- Blanton, J.L. & Wainwright, R.L. (1993). Multiple Vehicle Routing with Time and Capacity Constraint using Genetic Algorithms, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pp. 452-459, ISBN:1-55860-299-2, San Mateo, CA, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA
- Chakrabortya, U.K. (2008). Including Special Section: Genetic and Evolutionary Computing, *Information Sciences*, Vol. 178, N. 23, 4419-4420, ISSN: 0020-0255
- Chen, C.L. & Bulfin R.L. (1990). Scheduling unit processing time jobs on a single machine with multiple criteria. *Computers and Operations Research*, Vol. 17, N. 1, 1-7, ISSN: 0305-0548.
- Cook, S.A. (1971). The Complexity of Theorem Proving Procedures, *Proceedings of 3rd ACM Symposium on Theory of Computing*, pp. 151-158, Shaker Heights, Ohio, United States, May 03-05, 1971, ACM, New York, NY, USA
- Cook, S.A. (1983). An Overview of Computational Complexity. *Communications of the ACM*, Vol. 26, No.6, (June 1983) 400-408, ISSN: 0001-0782
- Cormen, H.T.; Leiserson,C.E.; Rivest, R.L. & Stein, C. (2001). *Introduction to Algorithms*, MIT Press, ISBN:0262032937, Cambridge, Massachusetts London, England
- Cruz-Chávez, M.A.; Díaz-Parra, O.; Juárez-Romero, D. & Martínez-Rangel, M.G. (2008). Memetic Algorithm Based on a Constraint Satisfaction Technique for VRPTW, In: *Artificial Intelligence and Soft Computing - ICAISC 2008*, Vol. 5097, Rutkowski, N L. et al. (Eds.), 376-387, Springer-Verlag, ISBN: 978-3-540-69572-1
- Garey, M. R. & Johnson, D. S. (1979). *Computers and Intractability, a Guide to the Theory of NP-completeness*, W. H. Freeman and Company, ISBN:0-7167-1044-7, New York
- Gładysz. B. (2007). Fuzzy robust courses scheduling problem, *Fuzzy Optimization and Decision Making*, Vol. 6, N. 2, 155-161, ISSN: 1568-4539
- Dahal, K.P.; Hossain,A.; Varghese,B.;Abraham,A.; Xhafa,F. & Daradoumis, A. (2008). Scheduling in Multiprocessor System Using Genetic Algorithms, *7th International Conference on Computer Information Systems and Industrial Management Applications (CISIM'08)*, pp. 277-282, ISBN 978-0-7695-3184-7, 2008, IEEE Computer Society press, USA.
- Holland, J.H. (1975). *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor
- Jain, L.C. ; Tan, S.C. & Lim, C.P. (2008). An Introduction to Computational Intelligence Paradigms. In: *Computational Intelligence Paradigms*, Vol. 137, Studies in Computational Intelligence. Lakhmi C. Jain et al. (Eds.), 1-23, Springer-Verlag. ISBN: 978-3-540-79473-8
- Jonsson, P. & Bäckström, C. (1995). Complexity Results for State-Variable Planning under Mixed Syntactical and Structural Restrictions, *Proceedings of the sixth international conference on Artificial intelligence: methodology, systems, applications*, pp. 205-213, ISBN:981-02-1877-X, Smolenice Castle, Slovakia, 1994, World Scientific Publishing Co., Inc. River Edge, NJ, USA
- Karp, R.M. (1972). Reducibility Among Combinatorial Problems, In: *Complexity of Computer Computations*, R.E. Miller and J.W. Thatcher, (Ed.), 85-104, Springer, ISBN: 0306307073

- Lim, C.P.; Jain, L.C.; Nguyen N.T. & Balas, V.E. (2008). Guest Editorial: Special issue on advances in computational intelligence paradigms and applications, *Fuzzy Optimization and Decision Making*, Vol. 7, N. 3, 215-217, ISSN 1568-4539
- McCabe, T.J. (1976). A complexity measure, *Transactions on Software Engineering*, Vol.2, N.4, (July 1976) 308-320, ISSN:0098-5589
- Mitchell, M. (1998). *An Introduction to genetic algorithms*, MIT Press, ISBN 0-262-13316-4, London, England
- Papadimitriou, C. & Steiglitz, K. (1982). *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, ISBN: 0-201-53082-1, New Jersey
- Pérez, J.; Pazos, R.A.; Frausto, J.; Rodríguez, G.; Romero, D. & Cruz, L. (2004). A Statistical Approach for Algorithm Selection, *Proceedings of III International Workshop on Experimental an Efficient Algorithms*, pp. 417-431, ISBN 3-540-22067-4, Angra dos Reis, Brazil, May 25-28, 2004, Springer-Verlag, Berlin
- Reinelt, G. (1991). TSPLIB - A Traveling Salesman Problem Library. *ORSA Journal on Computing*, Vol. 3, N. 4, 376-384, ISSN: 0899-1499
- Ruiz-Vanoye, J.A.; Díaz-Parra, O. & Landero, V. (2008). A Metric to Discriminate the Selection of Algorithms for General Problem ATSP, In: *Knowledge-Based Intelligent Information and Engineering Systems - KES 2008*, Vol. 5177, I. Lovrek, R.J. Howlett, and L.C. Jain (Eds.), 106-113, Springer-Verlag, ISBN: 978-3-540-85562-0, Berlin
- Ruiz-Vanoye, J.A.; Zárate, J.A.; Díaz-Parra, O. & Landero, V. (2008). Applied Statistical Indicators to the Vehicle Routing Problem with Time Windows for Discriminate Appropriately the Best Algorithm, In: *Computational Science and Its Applications - ICCSA 2008*, Vol. 5073, Part II, O. Gervasi et al. (Eds.), 1131-1141, Springer-Verlag, ISBN: 978-3-540-69840-1, Berlin
- Shannon, C.E. & Weaver, W. (1949). *The Mathematical Theory of Communication*, University of Illinois Press, ASIN: B000UFWID2, Urbana, Illinois
- Solomon, M.M. (1987). Algorithms for Vehicle Routing and Scheduling Problems with Time Window Constrains. *Operations Research*, Vol. 35, No. 2 (March-April 1987) 254-265, ISSN:0030-364X
- Toth, P. & Vigo, D. (2001). *The Vehicle Routing Problem*, Society for Industrial and Applied Mathematic (SIAM), ISBN: 0898714982, Philadelphia
- Thangiah, S.R. (1998). Genetic Algorithms, Tabu Search and Simulated Annealing Methods for Vehicle Routing Problems with Time Windows, In: *Practical Handbook of Genetic Algorithms: Complex Structures*, Vol. III: Complex structures, L. D. Chambers (Eds.), 347-381, CRC Press , ISBN: 0-8493-2539-0, Florida
- Wagner, S. & Affenzeller, M. (2004). The HeuristicLab Optimization Environment. *Technical Report*, Institute of Formal Models and Verification. Johannes Kepler University Linz. Austria
- Yamada, T. & Nakano, R. (1997). Genetic Algorithms for Job-Shop Scheduling Problems, *Proceedings of Modern Heuristic for Decision Support*, pp. 67-81, London, 18-19 March 1997, UNICOM seminar

Dependability Evaluation Based on System Monitoring

Janusz Sosnowski and Marcin Król

*Institute of Computer Science, Warsaw University of Technology
Poland*

1. Introduction

Recently dependability, maintainability and performance are becoming challenging issues for system designers and users. This results from the increasing complexity of hardware and software. In consequence these issues triggered various measurement-based studies in the literature, in particular they relate to the detection or prediction of critical situations. Most published results are focused on restricted and fragmented problems encountered in the systems or applications considered by the authors e.g. (Daniel et al., 2008; Ganapathi & Patterson, 2005; Hoffmann et al., 2007; Li et al., 2006; Makanju et al. 2008). In contemporary computer systems various monitoring mechanisms are provided, usually they relate to event and performance monitoring (John & Eckhout, 2006; Simache & Kaaniche 2001; Stearley, 2004; Zhang 2008; Ye, 2008). Such monitoring can generate a huge quantity of various data. An important issue is the selection and exploration of this data, to characterise the system operation. This is a non-trivial task even for experienced system administrators and analysts. Hence, it needs further investigations in the following aspects: system observation techniques, selecting the most sensitive observation parameters, creating the model of normal and abnormal (dangerous) behaviour of the system to facilitate problem identification and applying appropriate reactions.

In the literature most papers concentrate on finding some characteristic patterns in log files related to well defined critical problems encountered in the considered systems e.g. leading to system crash (Kalyanakrishnan et al., 1999; Sahoo et al., 2004; Xu et al., 1999). Various performance parameters have been monitored to predict specified network or processor bottlenecks (Cherkasova et al., 2008; Li et al., 2006; Reinders, 2007; Simache & Kaaniche, 2001), to detect attacks (Ye, 2008) or asses system dependability (Heath et al., 2002; Malek, 2008). Some statistical or data mining models have been developed for specific problems, however they are hardly applicable or irrelevant to other systems (Bertino et al., 1998; Hoffman et al., 2007, Lim et al., 2008). Hence, further studies covering various systems are still required to get better knowledge of monitoring capabilities and limitations.

We have faced many dependability and maintainability problems in computer systems used by students and scientists within the university for didactic and research purposes. Moreover, the load of the systems changes in time or place, hardware and software are updated or tuned, various maintenance and administrative activities occur sporadically, etc.

Hence, some operational problems or configuration inconsistencies arise. Such systems create a good basis for studying monitoring techniques. We have also some experience with other commercial systems handling many customers with fluctuating usage profiles. Long-term observations of these systems allowed us to improve monitoring techniques and dependability. For this purpose we have developed some special software modules, collected a lot of data and performed various analyses.

The paper outlines the main features of possible measurements (related to system operation) and the scope of collected data. On the basis of this survey we formulate problems of selecting and processing the collected data in relevance to dependability issues. We concentrate on software implemented monitoring systems, which provide combined exploration of event logs and performance counters. As opposed to other approaches the developed monitoring systems are interactive and adjusted to appearing problems. Moreover, we deal with a wider scope of observations, so we rely on many data sources simultaneously (e.g. event logs, performance logs, and exceptions). We have combined two approaches: identifying normal operation features and exploring long term trends (neglected in the literature); detecting various abnormalities. In both approaches we take into account correlation with environment and configuration changes. The paper describes this in relevance to two monitoring techniques based on various event logs and performance data. The presented considerations are illustrated with practical monitoring results. They relate to long term observations of many workstations and servers.

In section 2 we give an outline of event logs collected in computer systems. They are illustrated with some statistical data derived from long term observations of computers in didactic laboratories, some comments on data exploration are also included. Section 3 describes performance objects and related performance variables, which are usually monitored. The capabilities and problems with performance monitoring are presented in relevance to results from real systems. Final conclusions are given in section 4.

2. Event logs in computer systems

2.1 Event specifications and statistics

Computer systems are instrumented to provide various logs on their operation. These logs comprise huge amounts of data describing the status of system components, operational changes related to initiation or termination of services, configuration modifications, execution errors, etc. In Windows various events are stored in one of the three log files:

- *security log* comprises events related to system security and auditing processes,
- *system log* is used primarily to store diagnostic messages, abnormal conditions, events generated by system components (e.g. services, drivers),
- *application event log* reports errors that occur during the application execution (e.g. failing to allocate memory, aborting the transfer of a file, etc.).

Each event log record comprises the following fields:

- *event specification* - specifies 5 event types related to event severity level: error, warning, information, success audit, failure audit; this is supplemented with the event category, ID, date and time,
- *event source* - name of the user and the computer that generated the event,
- *description* - event details.

The list of possible events in Windows systems exceeds 10000 (Sosnowsk & Poleszak, 2006). In Unix and Linux systems over 10 sources of events and more priority levels are distinguished (*Syslog*). During normal operation of workstations or servers a large amount of events is registered in the logs. Hence, we have developed a special software system *LogMon* which collects data logs from specified computers within LAN and performs predefined processing to identify critical, abnormal and other situations (e.g. unavailability, warnings). *LogMon* co-operates with standard services (e.g. *Eventlog*) and provides some statistical and data exploration techniques. The performed analysis can be targeted at individual computers or specified computer subsets to find various correlations, etc.

The event files can be filtered preliminarily according to specified rules related to event identifier, source, type, system user, computer identifier, date and time (specified intervals by two points in time, specified month, week day, etc.). Complex multi step filtering is also possible, we can combine filtered files in one file, etc. Typical statistics relate to:

- 1) *Event counting* – the distribution (e.g. in decreasing order) of the number of registered events;
- 2) *Time between events* - time distribution between events of the same or different types;
- 3) *Event occurrence distribution* – statistics of the number of the selected event type in relevance to months, weeks, days or hours of the day;
- 4) *Event frequency profile* – the frequency of a selected event in the considered time period.

The calculated statistics are visualised in graphical forms, including a scatter plot where x axis is the time and y axis represents different event categories, or system components, etc. Such visualisations are useful to interpret the collected data, e.g. identification of significant patterns. The collected events can also be presented according to some ranking features e.g. frequency of appearance, entropy, etc.

The developed tool *LogMon* collects data from logs of many computers via internet. It provides many possibilities of filtering, searching specified event sequences, and visualising results. The log analysis can be targeted at different problems e.g. identifying critical situations, evaluating system availability, activity, system load, power problems. This process needs some knowledge of log specificity and experience with the used tool. We illustrate this in the subsequent section.

2.2 Illustrative results

While analysing the registered events we should identify the system start up and shutdown. When the Windows system is booted, event 6009 is logged and then it is followed by event 6005, which corresponds to *EventLog* service start-up. The termination of *EventLog* service registers event 6006. Event 6006 should be the last one registered in the system log after shutting down the system (clean shutdown). Nevertheless we observed some unexpected events registered after 6006 (anomalous situation). The event 6008 is recorded when a dirty shutdown (“blue screen”) occurs. The description part of this event comprises system time stamp (date and time). It may happen that the system cannot record 6006 or 6008 event, however 6009 and 6005 events are recorded. This complicates identification of system restarts, etc. After the system restart (e.g. in consequence of power supply outage) caused by event 6008 other registered events may give more details.

Within the events, which are correlated with system restarts, we can distinguish 4 groups: system and application updates, errors in applications and system services, hardware errors, unidentified restarts. Update events relate to restarts forced by installing new programs,

system updating or recovery of the previous version (with deletion of the updated ones). Typically they are initiated by: *Automatic Updates*, *NtService Pack*, *MsiInstaller*, *WindowsMedia*, *Print*. The events specify types of updates, information if it has been successfully accomplished or not, etc. For example for one of the computers the distribution of antivirus data base updates was as follows: for 270 registered events of this type (4570; *McUpdate*) 62 appeared in time period less than 1 day, 34 in the period from 1 to 2 days, etc. The analysis covered the log of 668 days. For some computers this frequency was sporadically disturbed – due to some configuration problems. Updates of different programs were performed successfully in over 80% cases, however for some computers non-successful updates were reported. The deeper analysis proved configuration inconsistency and network problems. Not successful clock synchronisation appeared on average in 10% cases.

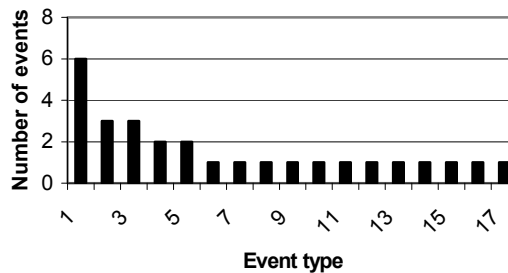


Fig. 1. Distribution of event types before restarts

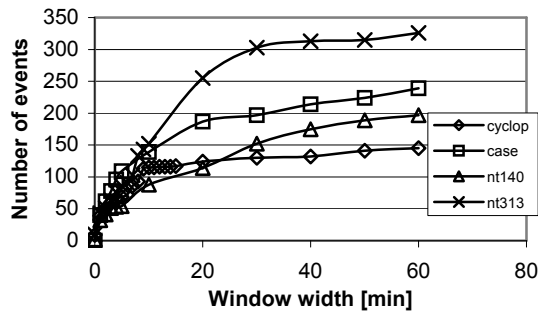


Fig. 2. The number of events within the specified time window

Looking for the sources of restarts we can analyse the distribution of events registered in the specified window before the event sequence related to reboot (6006, 6009, 6005) This is illustrated in fig. 1. The x-axis specifies different events In particular: $x=1$ relates to event *2013Srv* (the disk is almost full, you may need to delete some files); $x=2$ - event *54w32time* (the Windows Time Service was not able to find a Domain Controller, a time and date update was not possible); $x=13$ - event *21automatic program updates*; $x=15$ - event *26 application error*, etc. Such graph facilitates to identify the most frequent sources of restarts. Complete distribution

of all registered events in a decreasing order of occurrence is also useful to identify other problems. Typically 90% of registered events related to only 36 different event types (from the total list of over 10000 possible events).

To find sources of some event A it is useful to check events before this event within a specified time window Δ . For this purpose *LogMon* provides the capability of finding such statistics for a specified window Δ . Fig. 2 shows the number of registered events for 4 servers in function of the time window width Δ before restarts. We can observe some kind of saturation for $10 < \Delta < 30$ minutes, depending upon the system. Basing on the registered events we can identify restarts. For the considered servers (fig. 2) we have identified 24-60 restarts (on average 5 events per restart in the window).

The developed system *LogMon* provides various data exploration capabilities. In particular it can identify reasons of restarts and dirty restarts. In the case of restarts we have defined some regular expressions describing events most probably related to specified situations e.g. program update restarts. Tab. 1 shows restart statistics for 4 laboratories (L1-L4) each comprising 17 workstations and 3 servers (S1-S3). It gives the percent of restarts caused by program updates, application errors and hardware errors. In some cases the restart source is ambiguous - related to more than one source (mostly a program update and some other source). Quite significant percentage of detected restarts (unknown cause) did not comprise additional events facilitating their identification. They relate to power downs and restarts initiated by the user in response to some messages appearing on the screen, some of them can be identified from the application log. The table comprises the restart frequency (RF) expressed in the number of restarts per day (per single computer). Relatively low values of RF for two servers (S2 and S3) result from the stable profile of their usage.

	L1	L2	L3	L4	S1	S2	S3
RF	0.20	0.21	0.19	0.18	0.19	0.04	0.05
updates	20.2%	16.5%	22.6%	24.3%	32.5%	53.8%	3.9%
applic.	19.2%	29.7%	12.0%	29.6%	10.4%	15.4%	11.8%
hardware	1.2%	0.5%	2.0%	0.4%	9.1%	0%	0%
ambig.	1.4%	5.6%	5.0%	5.4%	6.5%	26.9%	1.3%
unknown	59.4%	53.4%	63.4%	45.7%	48.1%	30.8%	84.2%

Table 1. Restart statistics for workstations and servers

Special attention is needed to dirty restarts. Dirty restarts mostly relate to such events as 6008, 1000, 1001 *save dump*. Pressing RESET button also causes dirty restart. At the system level power down is treated in the same way as fast switching off the computer. In logs with power down closing event 6006 was missing. Analysing events in the window before the dirty restart we observed small number of events. This relates mostly to the situation with no possibility of recording the event due to the restart problem. In a period of 100 days we have identified typically 2-5 dirty restarts per computer. However, for a few computers this was in the range of 20-50 (old computers). In the case of servers practically the dirty restarts were caused by hardware errors. In the case of workstations dirty restarts were caused mostly by application errors, which caused system hang-ups or operational instability (due to developing and testing various program modules). Moreover, many student projects comprised critical errors leading to restarts.

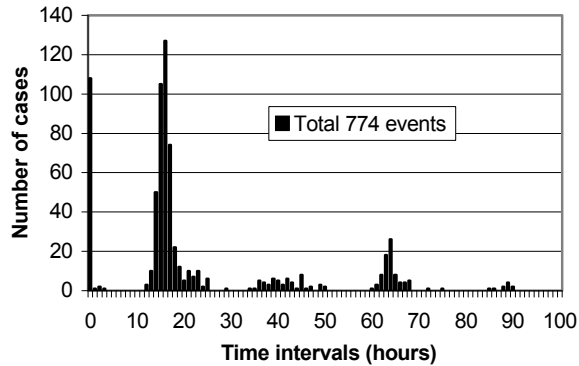


Fig. 3. Distribution of non activity periods in a workstation

Fig. 3 shows distribution of time periods between event A and B, which correspond to closing and starting the system. The x-axis of the plot has the granularity of single hours. The first bar (108 cases) relates to short time intervals (less than 1 hour) and it corresponds to short operation breaks related to restarts. The next group of higher bars relates to the periods of 15-16 hours, corresponding to switching off the computer for the evening and night periods. Subsequent groups of bars relate to longer non activity periods e.g. weekends, holidays, etc. Power supply problems can be directly identified by checking the time between events 3230:UPS (notification of power down and supply delivered from the batteries) and 3234:UPS (power recovery) or 3231:UPS (power switched off in the system) generated by UPS power supply. For an illustration we give some power statistics related to 2 servers SA and SB. In particular we specify the number of power down events per month for 9 subsequent months (October-June):

SA: 1; 1; 3; 1; 2; 2; 0; 0; 1 (total 9 events) and SB: 1; 9; 8; 7; 12; 10; 12; 1; 0 (total 60 events)

For all power down events the servers were supplied from UPS batteries, due to short power outages (power outages were tolerated by UPS). Power outages longer than 17 minutes and 150 sec result in SA and SB server switching off, respectively. The distribution of the duration of power outages was as follows: for server SA - (6 events) < 1 minute, 2 minutes ≤ (2 events) < 7 minutes and 1 event with 17 minutes duration; for server SB - (13 events) < 1s, 1s ≤ (33 events) < 2s, 2s ≤ (5 events) < 3 s, 3s ≤ (4 events) < 4s and 5s ≤ (4 events) ≤ 12s. Computers without UPS crashed and needed restarting. Some crashes appeared simultaneously in several computers (common power failure). In 3 cases the power downs signalled by UPSs of SA and SB servers were correlated (caused by the same power network outage), however the attributed timestamps differed by about 20 minutes, due to the lack of the clock synchronisation in both servers (such anomalies need identification).

An important issue is to evaluate the behaviour of various used programs within longer periods and correlate it with system upgrades, reconfiguration, load (number of users, processor stress). For an illustration table 2 shows the frequency distribution of registered program errors per day for two workstation (WS1 and WS2) within the one-year period. WS1 is less reliable due to higher number of used programs.

Computer	Number of errors per day							
	0	1	2	3	4	5	6	6
WS1	68%	10%	6%	3%	3%	2%	0%	1%
WS2	82%	5%	1%	2%	1%	2%	1%	0%

Table 2. Distribution of program errors for workstation WS1 and WS2

Events related directly to hardware faults appear before restarts. The most frequent relate to memory media, network cards, printers and other I/O devices. For example event 26 with description that the system could not write or read data from a specified file, device, etc. Other examples are: faulty block of CD ROM, timeout situation, IP address conflict, failure to load specified drivers, application errors, etc. It is worth noting that many events do not comprise descriptions, on the other hand some descriptions are ambiguous. Many faults can be identified from sequences of events. For this purpose some knowledge database can be systematically developed taking into account the gained experience from the system exploitation and maintenance.

2.3 Exploring data in event logs

Analysing logs is the basis for automatic system management and helpful in assuring high dependability. The registered reports may be related to different formats, the text messages are usually relatively short, contain a free format description of events (using a large size vocabulary), and quite often are ambiguous. Hence, data mining is not trivial and needs many preliminary studies to identify specificity and abnormal behaviour of the monitored systems. In this process some categorisation of text messages into a set of common classes over various system components is required. Moreover, an important issue is to visualise various statistics, temporal dependencies, etc.

Simple data mining can be targeted at discovering frequent and some specific well defined patterns (Lim et al., 2008; Makanju et al., 2008; Peng et al., 2005; Razavi & Kontogiannis, 2008; Vaarandi, 2008). In more advanced analysis of log reports we should take into account not only the individual messages but also their temporal dependencies, which can provide supplementary context information. For example a message on starting a program update may be followed by some errors due to inconsistency in the system configuration. Having transformed messages in some concise categorised form simplifies further data processing and finding characteristic patterns. Unfortunately, different systems use different log formats, etc. Hence, data collection and analysis has to be tuned to these systems. This is sufficient for individual system monitoring. In practice we are also interested in general properties of many systems, so some specification of similarities has to be defined to identify common characteristics, etc.

The log pre-processing may involve visualisation of event types or categories in relation to their appearance (time stamps). From such plot it is easy to identify some general properties e.g. the fact that event A usually happens after event B, the time distance between such events (it can be deterministic or random). An event may appear with some periodicity (e.g. antivirus updates, system heartbeat) or randomly. Some events may form a loop in a circular pattern or an event chain (e.g. related to a problem progress in predictable way). An event may appear simultaneously with other events. Various temporal relationships can be represented by appropriate graphs (Peng et al., 2007). Looking for temporal dependencies we analyse the distribution of time distance between events or compare the unconditional

probability of the waiting time for some event with a conditional probability in relevance to some other event. Various event patterns may signal system problems or confirm its health (e.g. heartbeats, successful program updates). Their interpretation can be simplified by correlating them with performance properties (section 3).

An important issue in tracing event logs is appropriate system configuration. If we are interested in monitoring system activity it is important to activate writing into logs supplementary information on user logins, file openings, processor usage, etc. It is also important to assure completeness of collected data. Some danger arises if logs are read periodically, so overwriting may happen. Hence, it is reasonable to collect this data systematically and storing it in a separate server. Complete information on all computers simplifies finding various correlations.

3. Performance Monitoring

3.1 Performance objects and variables

In most computer systems various data on performance can be collected in appropriate counters (e.g. provided by Windows, Linux) and according to some sampling policy (e.g. in 1-minute periods) (John & Eckhout, 2006; Reinders, 2007). These counters are correlated with performance objects such as processor, physical memory, cache, physical or logical discs, network interfaces, server of service programs (e.g. web services), I/O devices, etc. For each object many counters (variables) are defined characterising its operational state, usage, activities, abnormal behaviour, performance properties, etc. Special counters related to developed applications can also be added. These counters provide data useful for evaluating system dependability, predicting threats to undertake appropriate corrective actions, etc. The list of counters, which can be configured, is very long. For an illustration we describe some representative counters related to Windows systems.

Processor Time is the percentage of elapsed time that the processor spends to execute a non-idle thread. This counter is the primary indicator of processor activity, and displays the average percentage of busy time observed during the sample interval. *User Time* and *Privileged Time* relate to the percentage of elapsed time the processor spends in the user and in privileged mode, respectively. *Processor Queue Length* is the number of ready threads in the processor queue. *Processes* is the number of processes at the time of data collection. Similarly are counted threads, events, semaphores, etc. *Context Switches/sec* is the combined rate at which all processors on the computer are switched from one thread to another e.g. when a running thread voluntarily relinquishes the processor (is pre-empted by a higher priority ready thread), or switches between user-and privileged (kernel) mode to use an executive or subsystem service.

Interrupts/sec is the average rate, in incidents per second, at which the processor received and serviced hardware interrupts. It does not include deferred procedure calls (DPCs), which are counted separately. This value is an indirect indicator of the activity of devices that generate interrupts, such as the system clock, the mouse, disk drivers, network interface cards, and other peripheral devices. These devices normally interrupt the processor when they have completed a task or require attention. The system clock typically interrupts the processor every 10 milliseconds, creating a background of interrupt activity. This counter displays the difference between the values observed in the last two samples. *Interrupt Time*

is the time the processor spends receiving and servicing hardware interrupts during sample intervals.

System Up Time is the elapsed time (in seconds) that the computer has been running since it was last started till the current time. *C1 Time* is the percentage of time the processor spends in the C1 low-power idle state (enables the processor to maintain its entire context and quickly return to the running state), similar times are measured for C2 (a lower power and higher exit latency state than C1, it maintains the context of system cache) and C3 (a lower power and higher exit latency state than C2, is unable to maintain the coherency of its caches) states. There are also counters related to transitions to these states.

Available Bytes is the amount of physical memory, in bytes, available to processes running on the computer. It is calculated by adding the amount of space on the *Zeroed*, *Free*, and *Standby* memory lists. *Free* memory is ready for use; *Zeroed* memory consists of pages of memory filled with zeros to prevent subsequent processes from seeing data used by a previous process; *Standby* memory is memory that has been removed from a process working set (its physical memory) on route to disk, but is still available to be recalled. This counter displays the last observed value.

Free Space is the percentage of total usable space on the selected logical disk drive that was free. *Avg. Disk Bytes/Read* is the average number of bytes transferred from the disk during read operations, similar counter on write operations is available also.

Page Faults/sec is the average number of pages faulted per second (a referenced page in virtual memory is not available in the working area). Hard faults require disk access and soft faults cover faulted pages found elsewhere in physical memory. Most processors can handle large numbers of soft faults without significant consequences. However, hard faults, which require disk access, can cause significant delays. Similarly *Cache Faults/sec* is the rate at which faults occur when a page sought in the file system cache is not found and must be retrieved from elsewhere in memory or disk.

Page Reads/sec is the rate at which the disk was read (the number of read operations, without regard to the number of pages retrieved in each operation) to resolve hard page faults. *Pages Output/sec* is the rate at which pages are written to disk to free up space in physical memory. A high rate of *Pages Output* might indicate a memory shortage. *Pool Paged Failures* is the number of times allocations from paged pool have failed. It indicates that the computer's physical memory or paging file are too small.

File Read Operations/sec is the combined rate of file system read requests to all devices on the computer, including requests to read from the file system cache. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval. *File Control Operations/sec* is the combined rate of file system operations that are neither reads nor writes, such as file system control requests and requests for information about device characteristics or status. *Split IO/Sec* reports the rate at which I/Os to the disk were split into multiple I/Os. It may result from requesting data of a size that is too large to fit into a single I/O or that the disk is fragmented.

Server performance counters give: the number of bytes the server has received (or sent) from the network (indicates the server load); the number of sessions that have been closed due to unexpected error conditions or sessions that have reached the autodisconnect timeout and have been disconnected normally; failed logon attempts to the server (password guessing programs are being used to crack the security); the number of sessions that have been forced to logoff (due to logon time constraints); the number of sessions that have terminated

normally (this allows to find percentage of the sessions time outs or errors). Other counters provide some statistics on file operations such as: the number of times accesses to files opened successfully were denied (improper access authorisation, etc.), the number of failed file opens (attempting to access files not properly protected), the number of files currently opened in the server, the number of searches for files currently active, the number of sessions currently active in the server (indicates current server activity).

There are many counters characterising network traffic or TCP/IP protocol activity. Here are given some examples. *Bytes Received/sec* is the rate at which bytes are received over each network adapter, including framing characters. *Current Bandwidth* is an estimate of the current bandwidth of the network interface in bits per second. *Packets Received Errors* is the number of inbound packets that contained errors preventing delivery to a higher-layer protocol. *Packets Received Discarded* is the number of inbound packets that were discarded even though no errors had been detected (e.g. to free up buffer space). *Packets Received Unknown* is the number of packets received through the interface that were discarded because of an unknown or unsupported protocol. *Output Queue Length* is the length of the output packet queue, if this is longer than two, there are delays and the bottleneck should be found and eliminated. *Connection Failures* is the number of times TCP connections have made a direct transition to the CLOSED state from the SYN-SENT or SYN-RCVD state, and to the LISTEN state from the SYN-RCVD state.

There are also counters related to I/O devices e.g. for printers they count current number of jobs in a print queue, number of references (open handles) to this printer, peak number of references, current or maximal number of spooling jobs in a print queue. Accumulated statistics comprise data since the last restart e.g. number of out of paper errors, not ready errors and job errors in a print queue.

Resuming we can state that the number of possible performance variables is quite big and monitoring all of them is too expensive due to the additional load to the system processors and memory. Hence, an important issue is to select those variables which can provide the most useful information. This depends upon the goals of monitoring, the sensitivity of variables to the monitored properties of the system, the system usage profile, etc. To deal with this problem some preliminary studies of the system behaviour are needed. They facilitate tuning the monitoring tasks to the current needs and system specificity. We outline this problem in the next section.

3.2 Performance monitoring goals

Depending upon the goal of monitoring we have to select and configure appropriate counters within the objects of interest, to evaluate how well they are performing. Too large number of counters results in some additional load to the system and more complex data analysis. Hence, an important issue is to check which counters are most sensitive to the monitored problems. We have performed such studies in relevance to hardware and software failures as well as configuration or maintenance inconsistencies, effectiveness of some services, etc. Moreover, the applications can also use counter data to determine how much system resources to consume. For example, to determine how many data to transfer without competing for network bandwidth with other network traffic. The application could adjust its transfer rate as the bandwidth usage from other network traffic increases or decreases. Having specified performance counter thresholds we can generate alert

notifications, query performance data, create event tracing sessions, capture a computer's configuration, and trace the API calls in some of the Win32 system DLLs.

Most authors concentrate on well-defined critical problems e.g. cyberattacks or system availability. We have extended the scope of analysis to checking the normality of system operations e.g. periodicity of backups, program updates, acceptable level of signalled errors (e.g. rejected packets) and to detect abnormalities which may result in future problems, this relates mostly to long term observations and detecting dangerous trends e.g. decreasing of free memory. We correlate performance counters with event logs as well as with changes in configurations, system load, temporal disturbances in the operational environment (system maintenance and updates).

Some performance measures are directly used to balance system loads, etc. To assure this we have to analyse short term and long-term trends, correlate them with working hours, weekends, summer months (seasonal system behaviour), user activity profiles, etc. The considered systems were specific due to frequent configuration changes, many users with different profiles (students and different courses, projects, used programming environment, etc.) or servicing thousands of customers with random activities, influenced by various events (e.g. dynamic changes of the stock market).

Some problems are relatively easy to identify e.g. decreasing free memory in relevance to systematically increasing number of users and their higher engagement in more complex calculation problems, bigger data bases, etc. However, new not known problems are not evident and need deeper data multidimensional exploration. For example a higher rate of application warnings in the log was correlated with an installation of a new version of the operating system and the increased number of users. This related to configuration inconsistencies, which were alleviated later on.

Analysing the performance variables we can look at their instantaneous values, statistical properties, correlation with other variables or events. These statistics may relate to specified time periods. Moreover, we can target the analysis at averaged variable values (within specified periods, etc.) or analyse spikes, their frequencies, time distribution, periodicity, etc. All this depends upon the monitoring goal. For example in detection of cyber attacks we can try to find characteristic statistical deviations caused by the attack as compared with normal workload. Interesting studies have been presented in (Ye, 2008) for Windows systems. The authors give statistical properties of various performance variables related to different objects for 10 known cyber attacks. Analysing these results we have checked the observability properties of these attacks.

Property	Objects	Variables	Attacks	Sensitivity
M+	3-16 (7.7)	10-362 (105)	1-9 (7.7)	77/100
M-	3-11 (5.9)	17-182 (60)	1-10 (5.6)	59/180
DUL	1-4 (2.5)	1-33 (10.3)	1-8 (3.1)	25/80
DUR	3-9 (5.7)	9-52 (27.8)	1-10 (3.9)	58/110
DMM	3-9 (5.7)	24-52 (43.3)	2-9 (5.0)	57/120

Table 3. The impact of attacks on performance parameters

Tab. 3 shows minimal - maximal (average) numbers of monitored performance objects and related variables (counters) which reveal characteristic statistical properties during attacks. They related to an increase (M+) or decrease (M-) in the mean value, unimodal left skewed

(DUL), unimodal right skewed (DUR) and multimodal (DMM) distribution properties as compared with normal workload statistics. The 4-th table column gives the distribution (minimal, maximal and average) of the number of attacks affected by the considered objects (over each object class). The last column specifies the number of nonzero entries in the matrix correlating objects and attacks. Each entry in this matrix gives the number of object variables affecting (by specified statistical property) the appropriate attack. This gives some view on attack sensitivity (the number of all entries in the matrices is given after / character). For DUL, DUR and DMM statistics we observe lower number of affected objects and variables as compared with M+ and M-. For each distribution change we have found that *Process* object affects the most of attacks 7-10 (within all 10 attacks). Moreover, this object involves the biggest number of affected variables per single attack: up to 18, 21 and 23, for DUL, DUR and DMM distribution change, respectively. Some objects show maximal number of affected variables by specific attacks (dominating). Such sensitivity analysis allows the designer to minimise the number of monitored variables and assure good detection accuracy. For other problems we have to trace different properties, this is illustrated in the sequel.

3.3 Illustrative results

To give a better view on performance monitoring we present some illustrative results related to monitoring selected system objects and performance variables. The dynamic properties of these variables depend upon active applications, system load, environment interactions, etc. Hence, their behaviour in time can be very diversified resulting in different shapes and characteristics of related time plots. This creates some challenges for data exploration.

In general we can be interested in short term or long term monitoring results. In the first case we collect many samples which assure high accuracy. The results can be presented graphically with specified average (horizontal line), minimal and maximal values (vertical lines) for each sample. This is illustrated in fig 4a and 4b, which give the number of disc writes per second (y-axis covers the range 0-100 operations/s partitioned in 10 segments) for the system with no active application and for an application displaying a film of about 30 minutes (from a file in HDTV standard - 1280x720 pixels). The samples were collected every second. It is worth noting low activity of disc writes, nevertheless even in no active system there is some background activity related to operational system and Internet tasks. The both plots differ in time and amplitude distribution of spikes. Bigger difference was observed for processor usage (0% vs 24%) and disc read operations/s (no disc reads with 8 short spikes of 30 operations/s vs continuous average activity of 15 operations/s with many additional small spikes). The number of disc operations is much higher for disc defragmentation (on average 379 control operations per sec). Short-term observations are useful to find application properties, identify their disturbances etc. It is worth noting that the behaviour of performance variables may differ upon applications not only in the average values of analysed parameters (during the application run) but also in time (plot shapes). Quite often we observe some spikes in time plots of variables, their frequency and amplitudes may also characterise the applications (compare. fig. 4).

Long term observations give a view on general trends in the system. We illustrate this with some results in fig. 5-7. Fig. 5 shows some increase (from 20000 to over 100000) of transmitted bytes on the network in 6-month period. This resulted from adding new users.

Fig. 6 presents the number of created connections with a www server, the middle pulse (100-350 connections) corresponds to day hours 8.00-17.00, the negative pulse at the end of the plot corresponds to the switch problem on the next day (time period 8.00-10.00). Fig. 7 illustrates the number of connections related to 13 subsequent days. It is almost equal for all working days (a little bit over 200), much lower for Saturdays (below 100) and close to 0 for Sundays.

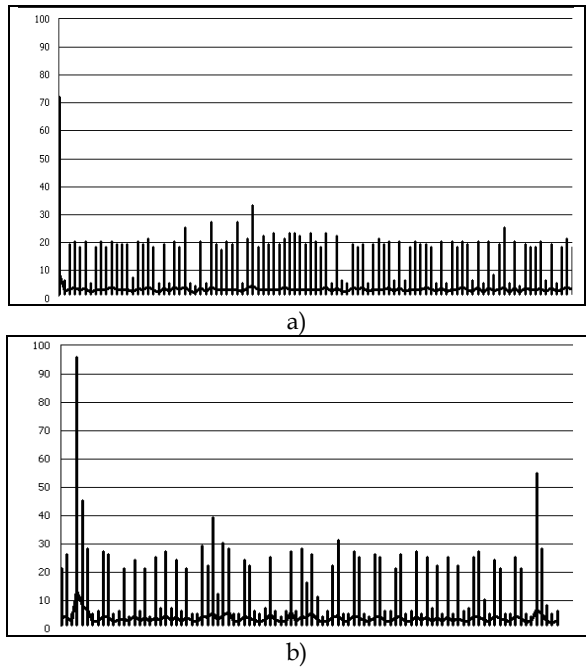


Fig. 4. Disc writes operations: a) no active applications, b) playing a film.

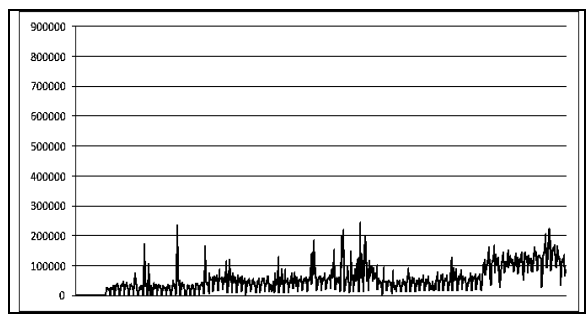


Fig. 5. Sent out bytes per second (half year profile)

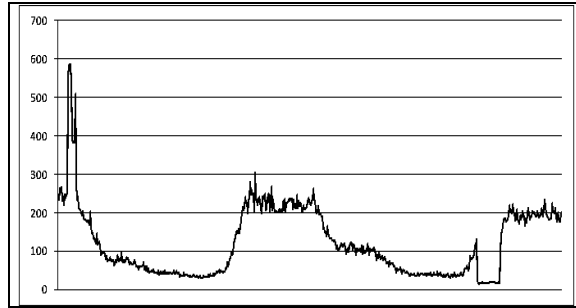


Fig. 6. The number of established connections in TCP (3 day profile)

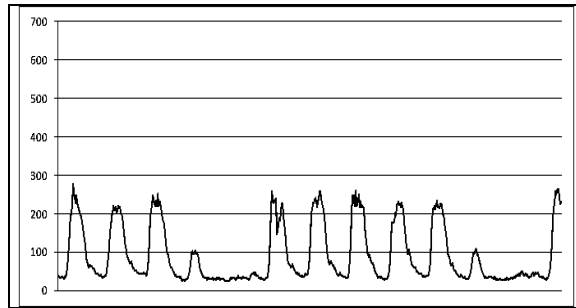


Fig. 7. The number of established connections in TCP (2 week profile)

Fig. 8 shows the number of active processes on Unix server *mion*, which handles Emails of students (about 3500 students within the Faculty of Electronics of our University) in the period 1st June till 31st December. The y-axis covers the range 0-25000 processes (partitioned into 5 equal segments – each 5500). The plot shows increasing trend of active processes. However on 12 September the system reconfiguration and restarting resulted in deleting many zombie and other not useful processes.

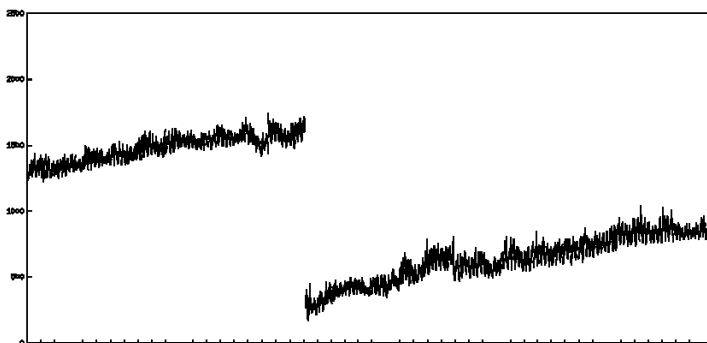


Fig. 8. The number of processes in the communication server (*mion*)

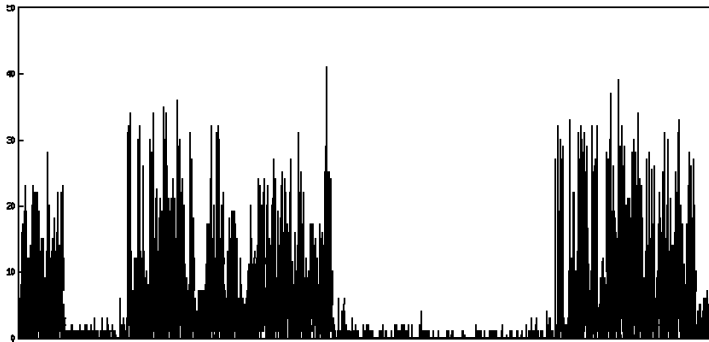


Fig. 9. The number of processes in the data processing server (*ikar*)

Fig. 9 shows the number of logged users to Unix server *ikar*, which handles programming laboratory with 16 workstations. The time scale on the x-axis covers the period of 12 months. The y-axis covers the range 0-50 users portioned in 5 equal segments (each 10 users). The first period of low activity relates to the winter vacation (February) and the second longer one corresponds to summer vacation (3.5 months).

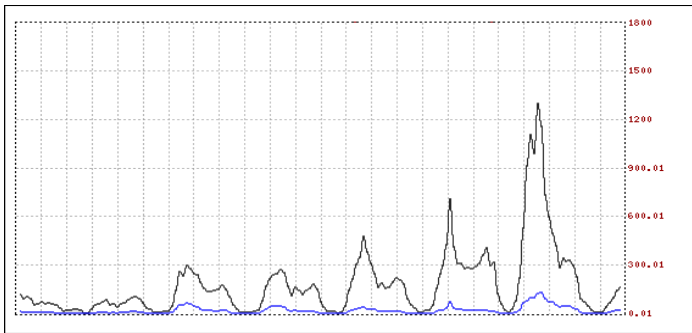


Fig. 10. Distribution of incoming session requests

Interesting observations were made for a farm of 16 servers providing some web services to many thousands of customers. The system uses quite sophisticated load balancing algorithm. Fig. 10 shows one-week plot of the total number of sessions (lower line relates to the number of sessions created within the last minute) handled by the farm within the 14th server. The y-axis covers the range 0-1800 session requests partitioned into 6 segments (300 requests each). The subsequent peaks of the plot relate to 7 days. The highest peak corresponds to Friday, for the weekend low activity is visible. In tab. 4 we give the server CPU usage (UP in percents) corresponding to the highest load on each day. Moreover we give also the outgoing traffic (OT in KB/sec) on the network port of the server. For each server the farm-managing program monitors its available resources (in particular processor and memory) and attributes user requests so as to achieve balanced load of all servers adapted to their functional capabilities and current activity. There are several classes of servers with different processors and architectural features. Hence, we take into account not

only the current values of performance parameters but also architectural capabilities. In particular the same level of processor usage expressed in percents does not reflect the real available processing power, which also depends upon the processor speed, etc. Monitoring the operation of all servers confirmed the effectiveness of the used load-balancing algorithm. Moreover, it allows finding critical deviations in farm operation and identifying problems (e.g. to eliminate a faulty server and move the traffic to other servers).

	Mon.	Tue.	Wed.	Thur.	Fri.	Sat.	Sun.
OT	700	600	500	550	620	150	30
UP	22%	23%	42%	38%	45%	8%	4%

Table 4. Outgoing traffic and processor usage in 14th server of a processing farm

The presented plots confirm a large diversity in possible shapes related both to normal and erroneous operation, hence their qualification needs advanced techniques, which take into account various correlation factors. In practice it is reasonable to correlate performance variables with event logs as well as environment changes, activities of administrators (maintenance, reconfigurations), software updates, user profiles, etc.

4. Conclusion

The available system logs and possible monitoring of various performance features provide enormous amount of data on system operation. This is a very useful source of information to evaluate and improve system dependability. However, selecting this information is not a trivial problem. It is possible to monitor and collect data on various aspects using pre-programmed counters, etc. Monitoring too many variables may result in system performance loss and high memory load with collected logs. So some optimisation is required here, in particular we can select the most sensitive variables related to various dependability issues. The next problem is interpretation of the collected data. This requires gaining some experience from long-term observations and correlating them with opinions of users and administrators. This simplifies creating procedures for automatic data exploration targeted at dependability issues. Hence, it is reasonable to enhance the available system mechanisms and software modules with an integrated database and advanced visualisation, statistical and data mining procedures (provided in the presented systems).

Further research is targeted at correlating various logs from many computers, identifying typical operational profiles, system loads, finding their changes in time and developing more efficient data exploration techniques to predict as soon as possible requirements of reconfigurations, detect inconsistencies or usage anomalies, etc. Having itemised specific patterns we can formulate appropriate actions. The gained experience is useful in defining event reduction rules, correlation rules (identifying events which are symptoms of specific problems), and problem avoidance rules (for some problems several stages of progress can be distinguished, early detection can prevent critical situation). We also plan to enhance the collected data from the field with logs relevant to injected faults (Sosnowski & Gawkowski 2006).

Acknowledgment

This work was supported by Ministry of Science and Higher Education grant 4297B/T02/2007/33. We express our appreciation to J. Machnicki for experiments related to fig. 7-10.

5. References

- Bertino, E., Ferrari, E. & Guerrini, G. (1998). An approach to model and query event based temporal data, *Proc. of 5th Int. Workshop on Temporal Representation and Reasoning*, pp. 122-131, IEEE Comp. Society
- Cherkasova, L.; Ozonar, K.; Mi, N.; Symons, J. & Smirni, E. (2008). Anomaly? application change? or workload change?, *Proc. of IEEE DSN Symposium*, pp. 452-461, IEEE Comp. Society, ISBN 1-4244-2398-9
- Daniel, E.; Lal, R. & Choi, G. (1999). Warnings and errors: A measurement study of a UNIX server, *FastAbstracts of IEEE Int. Symp. on Fault-Tolerant Computing*, FTCS-29, www.crhc.uiuc.edu/FTCS-29/fastabs.html.
- Ganapathi, A. & Patterson, D. (2005). Crash data collection: A windows case study, *Proc. of IEEE DSN Symposium*, pp.772-184, IEEE Comp. Society, ISBN 0-7695-2282-3
- Heath, T; Martin, R.P. & Nguyen, T.D. (2002). Improving cluster availability using workstation validation. *Proc. ACM SIG-METRICS Conf. Measurement and Modelling of Computer Systems*, pp.217-227.
- Hoffmann, G. A.; Trivedi, K.S. & Malek, M. (2007). A best practice guide to resource forecasting for computing systems, *IEEE Transactions on Reliability*, vol.56, no. 4, pp.615-628, ISSN 0018-9529
- John, L.K. & Eckhout, L, (editors) (2006). *Performance evaluation and benchmarking*, CRC Taylors & Francis, ISBN10-0-8493-3622-8
- Kalyanakrishman, M.; Kalbarczyk Z. & Iyer, R.K. (1999). Failure data analysis of a LAN of Windows NT based computers, *Proc. of 18th IEEE Symposium on Reliable Distributed Systems*, pp.178-188, IEEE Comp. Society
- Li, M.; Wang, S. & Zhao, W. (2006). A real-time and reliable approach to detecting traffic variations at abnormally high and low rates, In *ATC 2006, LNCS 4158*, 2006, L.T. Yang et al., (Ed.) pp.541-550, Springer Verlag, ISBN 3-540-69294-0, New York.
- Lim. Ch.; Singh, N. & Yainik, S. (2008). A log mining approach to failure analysis of enterprise telephony systems, *Proc. of IEEE DSN Symposium*, pp. 388-403, IEEE Comp. Society, ISBN 1-4244-2398-9
- Makanju A., Brooks, S.; Zincir-Heywood, A.N.& Milin, E.E.. (2008). LogView: visulizing event log clusters, *Proc. of Annual Conf. on Privacy, Security and Trust*, pp. 99-108, ISBN: 978-0-7695-3390-2
- Malek, M. (2008). Online dependability assessment through runtime monitoring and prediction, *Proc. of EDCC -7.*, pp.181, IEEE Comp. Society, ISBN 978-0-7695-3138-0
- Mansouri-Somani, M. & Sloman, H. (1996). A configurable event service for distributed systems, *Proc. of IEEE 3rd Int. Conf. on Configurable Distributed Systems*, pp. 210-217, IEEE Comp. Society, ISBN 0-8186-7395-8
- Peng, W.; Peng, Ch.; Li, T. & Wang, H. (2007). Event summarization for system management, *Proc. of ACM KDD'07*, pp. 1028-1032.

- Peng, W.; Li, T. & Ma, S. (2005). Mining logs files for computing system management, *SIGKDD Explorations*, vol. 7, issue 1, pp. 44-51.
- Razavi, A. & Kontogiannis, K. (2008). Pattern and policy driven log analysis for software monitoring, *Proc. of IEEE Int. Computer Software and Applications Conference*, pp.108-111, IEEE Comp. Society, ISSN 0730-3157
- Reinders, J. (2007). *VTune performance analyser essential*, Intel Press, ISBN0-9743649-5-9
- Sahoo, R.K.; Sivasubramanian, A.; Squillante, M. & Zhang, Y. (2004). Failure data analysis of a large-scale heterogeneous server environment, *Proc. of IEEE DSN Symposium*, pp.283-285, IEEE Comp. Society, ISBN 0-7695-2052-9
- Simache, C. & Kaàniche, M. (2002). Event Log Based Dependability Analysis of Windows NT and 2K Systems, *Proc. of IEEE Pacific Rim Int. Symposium on Dependable Computing*, pp.311-315, IEEE Comp. Society, ISBN 0-7695-1852-4
- Simache, C. & Kaàniche, M. (2001). Measurement-based availability of Unix systems in a distributed environment, *Proc. of 12th International Symposium on Software Reliability Engineering (ISSRE'01)*, pp.346-355, IEEE Comp. Society
- Sosnowski, J. & Gawkowski, P. (2006). Enhancing fault injection test bench, *Proc. of DepCos_RELCOMEX Conference*, pp.76-83, IEEE Comp. Society, ISBN 0-7695-2565-2
- Sosnowski, J. & Poleszak, M. (2006). On-line monitoring of computer systems, *Proc. of IEEE DELTA 2006 Workshop*. pp. 327-331, IEEE Comp. Society, ISBN 0-7695-2500-8, (complementary presentation in LATW 2006 and ICIT 2009)
- Stearley, J. (2004). Towards informatic analysis of Syslogs, *Proc. of IEEE Int. Conf. on Cluster Computing*, pp. 309-318, IEEE Comp. Society, ISBN 00-803-8430X
- Vaarandi, R. (2008). Mining event logs with SLCT and LogHound. *Proceedings of the 2008 IEEE/IFIP Network Operations and Management Symposium*, pp. 1071-1074.
- Xu, J.; Kallbarczyk, Z. & Iyer, R.K. (1999). Networked Windows NT system field failure data analysis. *Technical Report CRHC 9808*, University of Illinois at Urbana- Champaign,
- Ye, N. (2008). *Secure Computer and Network Systems*, John Wiley& Sons, Ltd. ISBN 978-0-470-02324-2, Chichester
- Zhang Y. & Sivasubramanian, A. (2008). Failure prediction IBM BlueGene?L event logs, *Proc. of Int. Symp. on Parallel and Distributed Processing*, pp.1-5, IEEE Comp. Society, ISBN 978-1-4244-2030-8